高機能 4 軸モーションコントロールIC

MCX514　取扱説明書

２０１４．　８．　１　　初版
２０１４．１２．１５　　２版
２０１５．　２．１０　　３版
２０１５．　４．１７　　４版
２０１５．１０．１５　　５版

NOVA electronics　　株式会社 ノヴァ電子

■改訂履歴

版 数	改訂年月日	改訂内容
初 版	2014年 8月 1日	新規作成
第2版	2014年12月15日	・以下誤記の修正 1.3 仕様一覧 オーバランリミット信号 端子名 3 補間 各補間速度 7.4.24 汎用入力値 読み出しのデータ範囲 9 制御プログラム例 補間命令関数の関数説明文
第3版	2015年 2月12日	・以下補間の終点範囲に関わる誤記を修正 3.1 直線補間 座標範囲 3.2 円弧補間 指定範囲 3.6 短軸パルス均一化 表 3.6.1 終点と円弧中心点の設定可能範囲 7.1 命令一覧 ■データ書き込み命令 ・移動パルス数／終点 設定のデータ範囲 ・円弧中心点 設定のデータ範囲 ・補間・終点最大値 設定のデータ範囲 ■データ読み出し命令 ・補間・終点最大値のデータ範囲 ・移動パルス数／終点のデータ範囲 7.2.7 移動パルス数／終点 設定のデータ範囲と終点座標 7.2.9 円弧中心点 設定のデータ範囲 7.2.26 補間・終点最大値 設定のデータ範囲と終点最大値 7.4.10 補間・終点最大値 読み出しのデータ範囲 7.4.22 移動パルス数／終点設定値 読み出しのデータ範囲 ・5.2 各信号名の説明:信号名 VDD の端子番号 誤記を修正
第4版	2015年4月17日	・以下三角防止機能に関わる誤記を修正 2.2.2 直線加減速の三角波形防止 2.2.3 非対称直線加減速の三角波形防止 ・以下補間ドライブ中のドライブ速度変更に関わる誤記を修正 3 章 補間の補間速度設定 3.7.1 連続補間の実施方法 3.7.4 連続補間の注意点 7.2.6 ドライブ速度設定 ・以下短軸パルス均一化機能に関わる誤記を修正、内容追記 3.6.2 短軸パルス均一化使用時の注意点 3.7.4 連続補間の注意点 ・以下誤記の修正 5.2 各信号の説明(D15～D0) 7.4.24 汎用入力値 読み出し
第5版	2015年10月15日	・以下 EMGN 信号入力信号に関わる追記 2.11.1 表 2.11-1※EMGN 信号の設定を追記 2.12.6 緊急停止について 4軸(全軸)を追記 5.2 各信号の説明 EMGN について 全軸を追記 ・以下の誤記を修正 10.1 周囲温度→動作温度

はじめに

このたびは、MCX514 をご検討いただき、ありがとうございます。

一般的に半導体製品は誤動作したり、故障する場合があります。本 IC をご使用いただく場合には、本 IC の誤動作や故障により人身・財産の損害が生じない様に、システムの安全設計をお願いします。

本 IC は一般電子機器（産業用自動化機器、産業用ロボット、計測機器、コンピュータ、事務機器、家電機器など）に使用されることを前提に作られています。特別に高い品質・信頼性が要求され、故障や誤動作が直接人命を脅かしたり人体に危害を及ぼす恐れのある機器（原子力制御機器、航空宇宙機器、輸送機器、医療機器、各種安全装置など）に使用されることを想定していませんし、動作の保証もされません。これらの高品質・高信頼性機器に使用することは、お客様の責任においてなされることになります。

輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、それらの法令の定めるところにより必要な手続きを行ってください。本 IC を大量破壊兵器の開発等の目的、軍事利用の目的、その他軍事用途の目的で使用しないでください。また、本 IC を国内外の法令及び規則により製造、使用、販売を禁止されている機器に使用することはできません。

本資料の掲載内容は技術進歩などにより予告なしに変更されることがあります。最新の資料を当社のホームページ（http://www.novaelec.co.jp）からダウンロードするか、当社に直接ご請求ください。

本資料に関するお問合せ、その他お気付きの点がございましたら、当社営業窓口までご連絡ください。

■ ご使用上の注意

本 IC のご使用につきましては、本マニュアルを十分にお読みいただいた上、信号電圧、信号タイミング、動作パラメータ値など記述された仕様範囲において、正しくご使用ください。

本 IC は、全てのモード、パラメータ値の組み合わせで動作を検証しておりません。お客様におかれましては、ご使用になるモード、パラメータ値において動作に問題がないかを十分に評価の上で使用してください。

IC 内部でプルアップされてない未使用入力端子の処置

使用しない入力端子は、必ず GND または VDD に接続してください。使用しない入力端子をオープン状態にしておくと、端子の信号レベルが不安定になり IC の機能不良の原因となります。
使用しない双方向端子は、必ず高抵抗(10k～100kΩ 程度)を介して VDD または GND に接続してください。直接 GND または VDD に接続すると、万一プログラムの誤りなどで出力状態になった場合には過電流が流れ IC が破損する場合があります。

リセットについて

本 IC の電源投入時には、必ず本 IC をリセットをする必要があります。安定したクロックが入力されている状態でクロック8サイクル以上の間 RESETN 信号を Low にすると本 IC はリセットされます。クロックが入力されていないと本 IC はリセットされませんのでご注意ください。

S字加減速ドライブの注意

本 IC は、定量パルスドライブを、加速／減速対称のS字加減速で自動的に減速・停止させる機能を持っています。しかし、初速度を極端に低く設定した場合には、多少の尻切れや引き摺りが発生する場合があります。S字加減速ドライブをご使用になられる場合には、お客様のシステムにおいて、この尻切れや引き摺りが許容できるか否かを十分に評価の上で使用してください。

■ 本書で使用する特殊用語・記号

アクティブ ある信号において、その信号の持つ機能が有効な状態であること。

ドライブ パルス列入力のサーボモータ、あるいはステッピングモータのドライバ(駆動装置)に対し、モータを回転させるためのパルスを出力する動作。

定量パルスドライブ 指定されたパルス量だけパルス出力するドライブ。相対位置ドライブ、反相対位置ドライブ、絶対位置ドライブの 3 種類が用意されています。

連続パルスドライブ 停止要因がアクティブになるまで無限にドライブパルスを出し続けるドライブ。

CW 時計方向(clockwiseの省略文字)。

CCW 反時計方向(counterclockwiseの省略文字)。

補間セグメント 連続補間を構成する1つ1つの補間ドライブ。

加速度増加率 単位時間当たりの加速度の増加／減少率。文字の表現は加速度の増加率だけですが、加速度の減少率も含めます。(=Jerk)

減速度増加率 単位時間当たりの減速度の増加／減少率。文字の表現は減速度の増加率だけですが、減速度の減少率も含めます。

2の補数 2進数における負の値の表現方法。(例)16 ビット長のデータでは、-1 は FFFFh、-2 は FFFEh、-3 は FFFDh、… -32768 は 8000h で表現します。

引き摺り 加減速定量パルスドライブの減速時において、初速度まで達してもまだ指定のドライブパルスを出し終えておらず、初速度で残りのドライブパルス出力する現象。(= Creep)

尻切れ 加減速定量パルスドライブの減速時において、初速度に達する前に指定のドライブパルスを出し終えてしまい、ドライブが終了してしまう現象。引き摺りの逆の現象。

↑ 信号が Low レベルから Hi レベルに変化するときの立ち上がりエッジ。

↓ 信号が Hi レベルから Low レベルに変化するときの立ち下がりエッジ。

n○○○○ X, Y, Z, Uの各軸の信号名をn○○○○と記述しています。この"n"はX, Y, ZおよびUを表します。

nPIOm X, Y, Z, Uの各軸のPIO信号をnPIOmと記述しています。この"n"はX, Y, ZおよびUを表し、"m"はPIO0～PIO7の0～7を表します。

SYNCm 同期動作セットSYNC0～SYNC3をSYNCmと記述しています。この"m"はSYNC0～SYNC3の0～3を表します。

MRm 多目的レジスタMR0～MR3をMRmと記述しています。この"m"はMR0～MR3の0～3を表します。

目　　次

１．　概要 ---------- 1

1.1 主な機能の特徴 ---------- 1
1.2 機能ブロック図 ---------- 9
1.3 仕様一覧 ---------- 11

２．　機能説明 ---------- 14

2.1 定量パルスドライブと連続パルスドライブ ---------- 14
2.1.1 相対位置ドライブ ---------- 14
2.1.2 絶対位置ドライブ ---------- 15
2.1.3 反相対位置ドライブ ---------- 15
2.1.4 連続パルスドライブ ---------- 17

2.2 加減速 ---------- 19
2.2.1 定速 ---------- 19
2.2.2 直線加減速（対称） ---------- 20
2.2.3 非対称直線加減速 ---------- 22
2.2.4 Ｓ字加減速（対称） ---------- 24
2.2.5 非対称Ｓ字加減速 ---------- 30
2.2.6 ドライブパルス幅と速度精度 ---------- 32

2.3 位置管理 ---------- 33
2.3.1 論理位置カウンタと実位置カウンタ ---------- 33
2.3.2 位置比較 ---------- 33
2.3.3 ソフトリミット ---------- 33
2.3.4 位置カウンタの可変リング ---------- 34

2.4 多目的レジスタ ---------- 35
2.4.1 比較対象と比較条件 ---------- 35
2.4.2 比較結果の用途（比較出力信号、同期動作、割り込み） ---------- 36
2.4.3 比較大小状態の表示 ---------- 39

2.5 自動原点出し ---------- 40
2.5.1 各ステップの動作 ---------- 41
2.5.2 偏差カウンタクリア出力 ---------- 44
2.5.3 ステップ間タイマー ---------- 44
2.5.4 サーチ速度とモードの設定 ---------- 45
2.5.5 自動原点出しの実行とステータス ---------- 49
2.5.6 自動原点出し時のエラー ---------- 50
2.5.7 自動原点出しの注意点 ---------- 51
2.5.8 自動原点出しの実例 ---------- 52

2.6 同期動作 ---------- 58
2.6.1 起動要因 ---------- 60
2.6.2 動作(Action) ---------- 62
2.6.3 同期動作の設定 ---------- 66
2.6.4 同期動作の実行 ---------- 69
2.6.5 同期動作による割り込み発生 ---------- 69
2.6.6 同期動作の実例 ---------- 70
2.6.7 同期動作の遅延時間 ---------- 75

2.7 スプリットパルス ---------- 77
2.7.1 スプリットパルスの設定 ---------- 77
2.7.2 スプリットパルスの開始／停止 ---------- 78
2.7.3 同期動作におけるスプリットパルス ---------- 79

2.7.4 スプリットパルスによる割り込み発生 ---- 79
2.7.5 スプリットパルスの注意点 ---- 79
2.7.6 スプリットパルスの実例 ---- 80

2.8 汎用入出力信号 ---- 86
2.8.1 nPIOm 信号 ---- 86
2.8.2 その他の入力信号 ---- 89

2.9 タイマー ---- 90
2.9.1 タイマーの動作 ---- 90
2.9.2 タイマーの設定 ---- 91
2.9.3 タイマーの始動と停止 ---- 91
2.9.4 タイマーと同期動作 ---- 91
2.9.5 タイマー動作状態と現在タイマー値の読み出し ---- 91
2.9.6 タイマーによる割り込み発生 ---- 91
2.9.7 タイマーの実例 ---- 92

2.10 割り込み ---- 95
2.10.1 X，Y，Z，U軸の割り込み ---- 95
2.10.2 連続補間の割り込み ---- 96

2.11 入力信号フィルタ ---- 97
2.11.1 入力信号フィルタ機能の設定 ---- 98
2.11.2 入力信号フィルタの設定例 ---- 99

2.12 その他の機能 ---- 100
2.12.1 外部信号によるドライブ操作（手動パルサー） ---- 100
2.12.2 ドライブパルス出力方式の選択 ---- 103
2.12.3 エンコーダ入力パルス方式の選択 ---- 104
2.12.4 ハードリミット信号 ---- 105
2.12.5 サーボモータドライバ対応の信号 ---- 106
2.12.6 緊急停止 ---- 106
2.12.7 ドライブ状態の出力 ---- 107

3. 補間 ---- 108

3.1 直線補間 ---- 110
3.1.1 終点最大値 ---- 110
3.1.2 直線補間の実例 ---- 110

3.2 円弧補間 ---- 112
3.2.1 終点判定 ---- 113
3.2.2 補間軸の入れ替え ---- 113
3.2.3 ＣW円弧補間ドライブの例 ---- 113

3.3 ヘリカル補間 ---- 114
3.3.1 補間軸の設定 ---- 115
3.3.2 補間速度データの設定 ---- 115
3.3.3 ヘリカル回転数の設定 ---- 115
3.3.4 位置データの設定 ---- 116
3.3.5 ヘリカル演算の実行 ---- 117
3.3.6 ヘリカル補間の実行 ---- 118
3.3.7 現在ヘリカル回転数の読み出し ---- 118
3.3.8 ヘリカル補間の位置変動 ---- 119
3.3.9 ヘリカル補間の実例 ---- 120

3.4 ビットパターン補間 ---- 123
3.4.1 補間軸指定 ---- 124
3.4.2 補間速度設定 ---- 124

3.4.3 ビットパターンデータの書き込み ---------- 124
3.4.4 補間ドライブ命令の発行 ---------- 125
3.4.5 補間終了 ---------- 125
3.4.6 プリバッファの空きを確認 ---------- 126
3.4.7 補間ドライブの中断 ---------- 126
3.4.8 ビットパターン補間の実例 ---------- 127

3.5 線速一定 ---------- 128
3.5.1 線速一定の設定 ---------- 129

3.6 短軸パルス均一化 ---------- 130
3.6.1 短軸パルス均一化の設定 ---------- 130
3.6.2 短軸パルス均一化使用時の注意点 ---------- 131

3.7 連続補間 ---------- 132
3.7.1 連続補間の実施方法 ---------- 133
3.7.2 割り込みを用いた連続補間 ---------- 135
3.7.3 連続補間時のエラー発生 ---------- 136
3.7.4 連続補間の注意点 ---------- 136
3.7.5 連続補間の実例 ---------- 137

3.8 加減速ドライブでの補間 ---------- 139
3.8.1 直線補間の加減速度ライブ ---------- 139
3.8.2 円弧補間、ビットパターン補間の加減速ドライブ ---------- 139
3.8.3 連続補間の加減速ドライブ ---------- 141

3.9 補間ステップ送り ---------- 142
3.9.1 コマンドによる補間ステップ送り ---------- 142
3.9.2 外部信号による補間ステップ送り ---------- 143
3.9.3 補間ステップ送りの注意点 ---------- 143

3.10 マルチチップ補間 ---------- 144
3.10.1 実行手順 ---------- 145
3.10.2 補間ドライブの途中停止 ---------- 147
3.10.3 連続補間 ---------- 147
3.10.4 マルチチップ補間の注意点 ---------- 147
3.10.5 マルチチップ補間の実例 ---------- 148

4. I^2C シリアルバス ---------- 152

4.1 I^2C バスモードで使用する端子 ---------- 152
4.1.1 プルアップ抵抗(Rp) ---------- 152
4.1.2 I2CRSTN リセット ---------- 153
4.2 I^2C バス送受信手順 ---------- 153
4.2.1 書き込み動作 ---------- 154
4.2.2 読み出し動作 ---------- 155
4.2.3 I^2C シリアルバス使用時の注意点 ---------- 157
4.2.4 接続例 ---------- 157
4.2.5 制御例 ---------- 158

5. 端子配置と各信号の説明 ---------- 161

5.1 端子配置 ---------- 161
5.2 各信号の説明 ---------- 162
5.3 入／出力回路 ---------- 167
5.4 回路設計上の注意 ---------- 168

６． リード／ライトレジスタ ---------- 169

6.1 16 ビットデータバスのレジスタアドレス ---------- 169
6.2 8 ビットデータバスのレジスタアドレス ---------- 171
6.3 I²C シリアルインターフェイスバスモードのレジスタアドレス ---------- 171
6.4 ＷＲ０ コマンドレジスタ ---------- 172
6.5 ＷＲ１ モードレジスタ１ ---------- 172
6.6 ＷＲ２ モードレジスタ２ ---------- 173
6.7 ＷＲ３ モードレジスタ３ ---------- 174
6.8 ＷＲ４ アウトプットレジスタ１ ---------- 177
6.9 ＷＲ５ アウトプットレジスタ２ ---------- 177
6.10 ＷＲ６，７ ライトデータレジスタ１，２ ---------- 177
6.11 ＲＲ０ 主ステータスレジスタ ---------- 178
6.12 ＲＲ１ ステータスレジスタ１ ---------- 179
6.13 ＲＲ２ ステータスレジスタ２ ---------- 180
6.14 ＲＲ３ ステータスレジスタ３ ---------- 181
6.15 ＲＲ４ ＰＩＯリードレジスタ１ ---------- 183
6.16 ＲＲ５ ＰＩＯリードレジスタ２ ---------- 183
6.17 ＲＲ６，７ リードデータレジスタ１，２ ---------- 183

７． 命令 ---------- 184

7.1 命令一覧 ---------- 184

7.2 データ書き込み命令 ---------- 188
7.2.1 加速度増加率 設定 ＪＫ ---------- 188
7.2.2 減速度増加率 設定 ＤＪ ---------- 188
7.2.3 加速度 設定 ＡＣ ---------- 189
7.2.4 減速度 設定 ＤＣ ---------- 189
7.2.5 初速度 設定 ＳＶ ---------- 190
7.2.6 ドライブ速度 設定 ＤＶ ---------- 190
7.2.7 移動パルス数／終点 設定 ＴＰ ---------- 191
7.2.8 マニュアル減速点 設定 ＤＰ ---------- 191
7.2.9 円弧中心点 設定 ＣＴ ---------- 192
7.2.10 論理位置カウンタ 設定 ＬＰ ---------- 192
7.2.11 実位置カウンタ 設定 ＲＰ ---------- 192
7.2.12 ソフトリミット＋ 設定 ＳＰ ---------- 192
7.2.13 ソフトリミットー 設定 ＳＭ ---------- 193
7.2.14 加速カウンタオフセット 設定 ＡＯ ---------- 193
7.2.15 論理位置カウンタ最大値 設定 ＬＸ ---------- 193
7.2.16 実位置カウンタ最大値 設定 ＲＸ ---------- 193
7.2.17 多目的レジスタ０ 設定 ＭＲ０ ---------- 194
7.2.18 多目的レジスタ１ 設定 ＭＲ１ ---------- 194
7.2.19 多目的レジスタ２ 設定 ＭＲ２ ---------- 194
7.2.20 多目的レジスタ３ 設定 ＭＲ３ ---------- 195
7.2.21 原点検出速度 設定 ＨＶ ---------- 195
7.2.22 速度増減値 設定 ＩＶ ---------- 195
7.2.23 タイマー値 設定 ＴＭ ---------- 196
7.2.24 スプリットパルス設定１ ＳＰ１ ---------- 196
7.2.25 スプリットパルス設定２ ＳＰ２ ---------- 196
7.2.26 補間･終点最大値 設定 ＴＸ ---------- 197
7.2.27 ヘリカル回転数 設定 ＨＮ ---------- 197
7.2.28 ヘリカル演算値 設定 ＨＶ ---------- 197

7.3 モード書き込み命令 ---------- 198
7.3.1 多目的レジスタモード 設定 ＭＲＭ ---------- 198
7.3.2 ＰＩＯ信号設定１ Ｐ１Ｍ ---------- 199
7.3.3 ＰＩＯ信号設定２・その他設定 Ｐ２Ｍ ---------- 201
7.3.4 自動原点出しモード設定１ Ｈ１Ｍ ---------- 202

7.3.5 自動原点出しモード設定２ Ｈ２Ｍ ---------- 203
7.3.6 入力信号フィルタモード 設定 ＦＬＭ ---------- 205
7.3.7 同期動作ＳＹＮＣ０，１，２，３ 設定
Ｓ０Ｍ，Ｓ１Ｍ，Ｓ２Ｍ，Ｓ３Ｍ ---------- 206
7.3.8 補間モード 設定 ＩＰＭ ---------- 208

7.4 データ読み出し命令 ---------- 210
7.4.1 論理位置カウンタ 読み出し ＬＰ ---------- 210
7.4.2 実位置カウンタ 読み出し ＲＰ ---------- 210
7.4.3 現在ドライブ速度 読み出し ＣＶ ---------- 210
7.4.4 現在加減速度 読み出し ＣＡ ---------- 211
7.4.5 多目的レジスタ０ 読み出し ＭＲ０ ---------- 211
7.4.6 多目的レジスタ１ 読み出し ＭＲ１ ---------- 211
7.4.7 多目的レジスタ２ 読み出し ＭＲ２ ---------- 211
7.4.8 多目的レジスタ３ 読み出し ＭＲ３ ---------- 212
7.4.9 現在タイマー値 読み出し ＣＴ ---------- 212
7.4.10 補間･終点最大値 読み出し ＴＸ ---------- 212
7.4.11 現在ヘリカル回転数 読み出し ＣＨ ---------- 212
7.4.12 ヘリカル演算値 読み出し ＨＶ ---------- 213
7.4.13 ＷＲ１設定値 読み出し ＷＲ１ ---------- 213
7.4.14 ＷＲ２設定値 読み出し ＷＲ２ ---------- 213
7.4.15 ＷＲ３設定値 読み出し ＷＲ３ ---------- 213
7.4.16 多目的レジスタモード設定 読み出し ＭＲＭ ---------- 214
7.4.17 ＰＩＯ信号設定１ 読み出し Ｐ１Ｍ ---------- 214
7.4.18 ＰＩＯ信号設定２・その他設定 読み出し
Ｐ２Ｍ ---------- 214
7.4.19 加速度設定値 読み出し ＡＣ ---------- 215
7.4.20 初速度設定値 読み出し ＳＶ ---------- 215
7.4.21 ドライブ速度設定値 読み出し ＤＶ ---------- 215
7.4.22 移動パルス数／終点設定値 読み出し ＴＰ ---------- 215
7.4.23 スプリットパルス設定１ 読み出し ＳＰ１ ---------- 216
7.4.24 汎用入力値 読み出し ＵＩ ---------- 216

7.5 ドライブ命令 ---------- 217
7.5.1 相対位置ドライブ ---------- 217
7.5.2 反相対位置ドライブ ---------- 218
7.5.3 ＋方向連続パルスドライブ ---------- 218
7.5.4 －方向連続パルスドライブ ---------- 218
7.5.5 絶対位置ドライブ ---------- 219
7.5.6 ドライブ減速停止 ---------- 219
7.5.7 ドライブ即停止 ---------- 219
7.5.8 方向信号＋設定 ---------- 219
7.5.9 方向信号－設定 ---------- 220
7.5.10 自動原点出し実行 ---------- 220

7.6 補間命令 ---------- 221
7.6.1 １軸直線補間ドライブ(マルチチップ) ---------- 221
7.6.2 ２軸直線補間ドライブ ---------- 221
7.6.3 ３軸直線補間ドライブ ---------- 221
7.6.4 ４軸直線補間ドライブ ---------- 222
7.6.5 ＣＷ円弧補間ドライブ ---------- 222
7.6.6 ＣＣＷ円弧補間ドライブ ---------- 222
7.6.7 ２軸ビットパターン補間ドライブ ---------- 222
7.6.8 ３軸ビットパターン補間ドライブ ---------- 223
7.6.9 ４軸ビットパターン補間ドライブ ---------- 223
7.6.10 ＣＷヘリカル補間ドライブ ---------- 223
7.6.11 ＣＣＷヘリカル補間ドライブ ---------- 223
7.6.12 ＣＷヘリカル演算 ---------- 224
7.6.13 ＣＣＷヘリカル演算 ---------- 224

7.6.14 減速有効 ---------- 224
7.6.15 減速無効 ---------- 225
7.6.16 補間割り込みクリア／補間ステップ ---------- 225

7.7 同期動作操作命令 ---------- 226
7.7.1 同期動作 有効設定 ---------- 226
7.7.2 同期動作 無効設定 ---------- 227
7.7.3 同期動作 起動 ---------- 227

7.8 その他の命令 ---------- 228
7.8.1 速度増加 ---------- 228
7.8.2 速度減少 ---------- 228
7.8.3 偏差カウンタクリア出力 ---------- 229
7.8.4 タイマー始動 ---------- 229
7.8.5 タイマー停止 ---------- 229
7.8.6 スプリットパルス開始 ---------- 229
7.8.7 スプリットパルス停止 ---------- 230
7.8.8 ドライブ開始ホールド ---------- 230
7.8.9 ドライブ開始フリー ---------- 230
7.8.10 エラー・終了ステータスクリア ---------- 230
7.8.11 RR３ ページ０表示 ---------- 231
7.8.12 RR３ ページ１表示 ---------- 231
7.8.13 終点最大値クリア ---------- 231
7.8.14 ＮＯＰ ---------- 231
7.8.15 コマンドリセット ---------- 232

８． 入出力信号接続例 ---------- 233

8.1 １６ビット／８ビットバスモードの接続例 ---------- 233
8.2 Ｉ²Ｃバスモードの接続例 ---------- 234
8.3 モーションシステム構成例 ---------- 235
8.4 ドライブパルス出力回路例 ---------- 235
8.5 リミット等入力信号の接続例 ---------- 236
8.6 エンコーダ入力信号の接続例 ---------- 236

９． 制御プログラム例 ---------- 237

１０． 電気的特性 ---------- 249

10.1 ＤＣ特性 ---------- 249
10.2 ＡＣ遅延特性 ---------- 250
10.2.1 クロック ---------- 250
10.2.2 ＣＰＵリード／ライトサイクル ---------- 250
10.2.3 ＣＬＫ／出力遅延特性 ---------- 251
10.2.4 入力パルス ---------- 251
10.2.5 汎用入／出力信号（ｎＰＩＯ７～０） ---------- 252
10.2.6 スプリットパルス ---------- 252
10.2.7 Ｉ²Ｃシリアルバス ---------- 253

１１． 入出力信号タイミング ---------- 254

11.1 パワーオンタイミング ---------- 254
11.2 ドライブ開始／終了時 ---------- 254
11.3 補間ドライブ開始 ---------- 255
11.4 ドライブ開始フリー ---------- 255
11.5 ドライブ即停止 ---------- 255
11.6 ドライブ減速停止 ---------- 256
11.7 スプリットパルスの詳細タイミング ---------- 256

１２．　外形寸法 ---------- 257

１３．　保管条件と推奨実装条件 ---------- 258

13.1 本ＩＣの保管について ---------- 258
13.2 はんだごてによる標準実装条件 ---------- 258
13.3 リフローによる標準実装条件 ---------- 258

付録 ----------

付録 A　加減速ドライブのための計算式 ---------- A-1
A-1　直線加減速ドライブの場合 ---------- A-1
A-2　S 字加減速ドライブの場合 ---------- A-2
付録 B　入力クロックが 16MHz 以外のパラメータ計算式 ---------- B-1
付録 C　MCX300 シリーズとの相違点 ---------- C-1

1. 概要

1.1 主な機能の特徴

MCX514 は従来品の MCX314As/MCX314AL を大幅に機能アップした 4 軸モーションコントロール IC です。

補間機能については、従来の直線補間、円弧補間、ビットパターン補間に加えて、XY平面上の円弧補間と同期してZ軸を垂直方向に移動させるヘリカル補間機能を装備しています。

MCX500シリーズのモーションコントロール IC は、速度倍率(レンジ)設定をなくしました。これにより、速度制御ドライブにおいては、1pps の超低速から最高 8Mpps までドライブ速度をリニアに(レンジ切り替えなしに)可変でき、かつ 1pps 単位で自由に出力することができます。

上位 CPU とのインターフェイスは、従来の 16 ビット/8 ビットバスの他に I^2C シリアルインターフェイスバスを装備しています。パラレルバスを持っていない CPU との接続も可能です。

■ ヘリカル補間

MCX514 は、従来の直線・円弧補間に加え、ヘリカル補間ドライブを実行させることが可能です。
ヘリカル補間は、XY平面(直交座標)における円弧補間ドライブに同期して他の軸を移動させる動作です。下の図はXY平面の円弧補間に合わせて、Z軸を＋方向に移動させている例です。図 1.1-1 a.は円弧補間 1 周以内のヘリカル補間を示し、図 1.1-1 b.は複数回転の例を示しています。MCX514 ではこれら両方の補間を行わせることができます。

a. 1 周以内　　b. ヘリカル補間の例(1 周以上)

図 1.1-1 ヘリカル補間の例

またヘリカル補間の応用例として、XY平面上の円弧補間に合わせて他の回転軸を一定の回転角で回転させる、円弧上の法線制御動作を行うことができます。図 1.1-2 はXY平面上で円弧補間を行う台座に回転軸を設けて、台座上のカメラやノズルなどが常に円弧補間の中心を向くように動作する例を表しています。

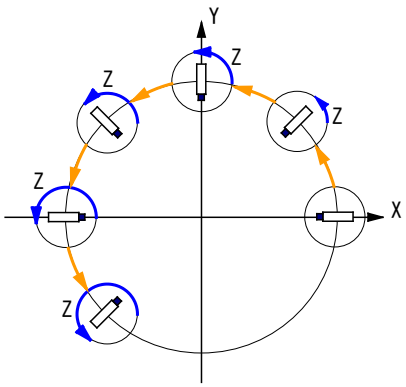


図 1.1-2 ＸＹ軸円弧補間とＺ軸の法線制御の例

■ 連続補間のための 8 段プリバッファ

MCX514 は高速の連続補間ドライブに対応するために、各セグメントの終点データ(その他)を蓄える 8 段のプリバッファレジスタを備えています。
従来の 1 段のプリバッファしか持たない MCX314A において連続補間を行う場合は、各補間セグメントの移動時間が次セグメント位置データセット時間より必ず長くなければなりませんでした。そのために各セグメントの最小移動パルス量は補間ドライブ速度によって制約されます。例えば、CPU のデータセット時間 T_{DS}=80 μsec、補間ドライブ速度 V=100Kpps であれば最小移動パルス量は 8 パルス以上必要であり、これより少ない移動パルスのセグメントは実行することができません。

MCX514 では、プリバッファを 8 段に増設し、この制約を大幅に改善することができます。右の図に示すような連続補間を行う場合、Seg3 のような短いセグメントが存在しても、Seg3 を含む8個のセグメントの平均移動時間が次セグメント位置データセット時間より長ければ、連続補間を行うことができます。

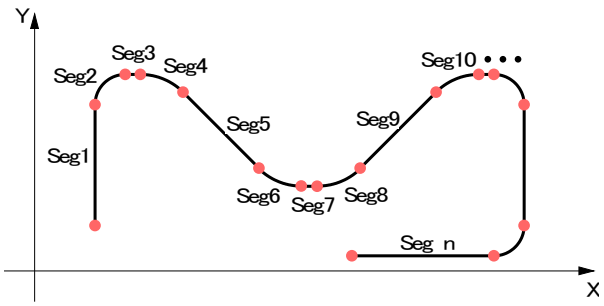


図 1.1-3 連続補間の例

■ マルチチップ補間

MCX514 を複数チップ連結して 5 軸以上の多軸直線補間を行うことができます。各チップは 8 本のマルチチップ信号線で並列に接続します。
多軸直線補間では補間を行うすべての軸の終点に対する最大値が補間演算で必要になりますが、MCX514 ではこの最大値をセットする必要はありません。上位 CPU が各軸の終点データをそれぞれ IC に対して書き込むと、そのデータはマルチチップ信号線を経由して各 IC に送信され、終点最大値は IC 内で自動的に生成されます。

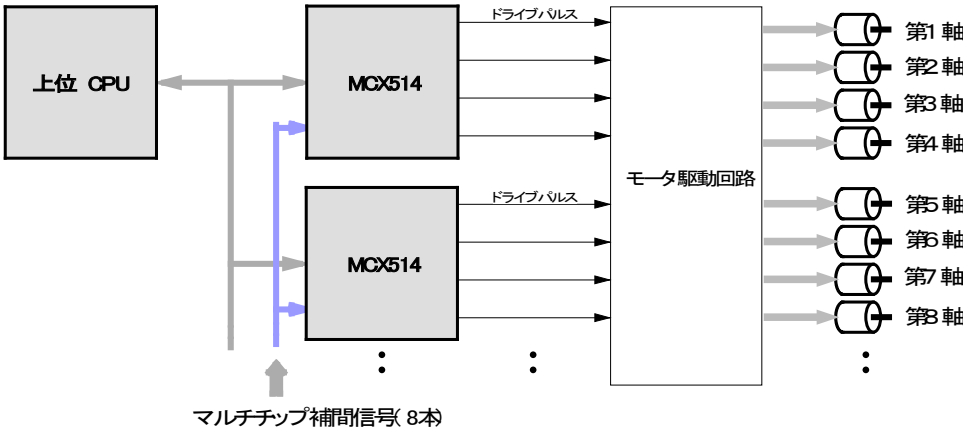


図 1.1-4 マルチチップ補間の例

■ 補間の短軸パルス均一モード

補間ドライブでは、補間を行うすべての軸がドライブ中常に均一の周期でドライブパルスを出力するわけではありません。例えば下の図に示すように、2 軸直線補間において、移動量(パルス)が大きい方の軸(長軸)は常にパルスを出力し続けますが、少ない方の軸(短軸)は補間演算結果によりパルスを出力するときとしないときがあります。この間引きが問題になる事があります。ステッピングモータで補間を行う場合に、独立ドライブと同じように高速度で補間を行おうとすると、短軸はこの間引きパルスのために振動が増加し、脱調する場合があります。MCX514 では、短軸パルス均一化機能によりこの問題を改善することができます。移動量が少ない軸においても、極力パルス周期を均一に近づけてドライブパルスを出力します。 線速一定モードと併用すると線速一定の精度が向上します。

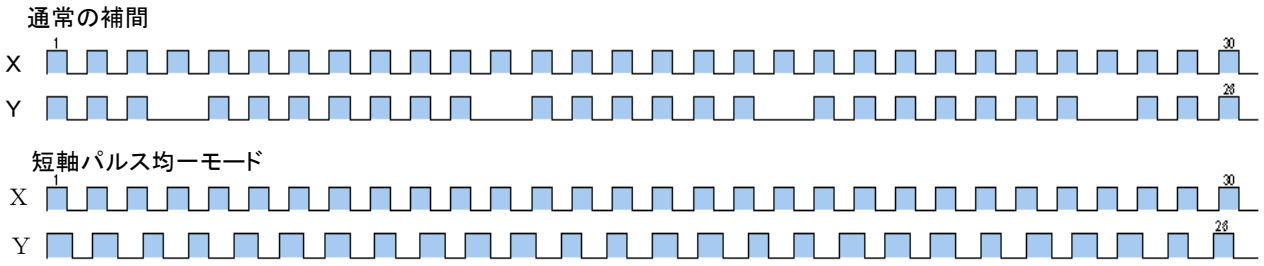


図 1.1-5 移動量 X 30 パルス Y 26 パルスの２軸直線補間時のドライブパルス出力

■ 2軸高精度線速一定モード

線速とは、補間ドライブを行う時の軌跡先端の移動速度のことで、ヘッドスピードとも言います。補間ドライブを行いながらワークの加工を行う、塗布するなどの動作では、この線速を一定に保つことが重要な機能になります。
本 IC では、従来の線速一定モードに加えて、大幅に線速一定精度を向上させた2軸高精度線速一定モードを実現しています。2軸直線補間、円弧補間、ヘリカル補間ドライブ時において、前述の短軸パルス均一モードと2軸高精度線速一定モードを組み合わせて使った場合、線速の速度偏差を±0.2%以下に収めることができ、補間ドライブ時の大幅な速度の精度向上が期待できます。
下の図は、半径 10,000 パルスの円弧補間ドライブを、従来の線速一定モードで行った時と、MCX514 の2軸高精度線速一定モードで行った時の、それぞれの速度偏差をグラフで表しています。

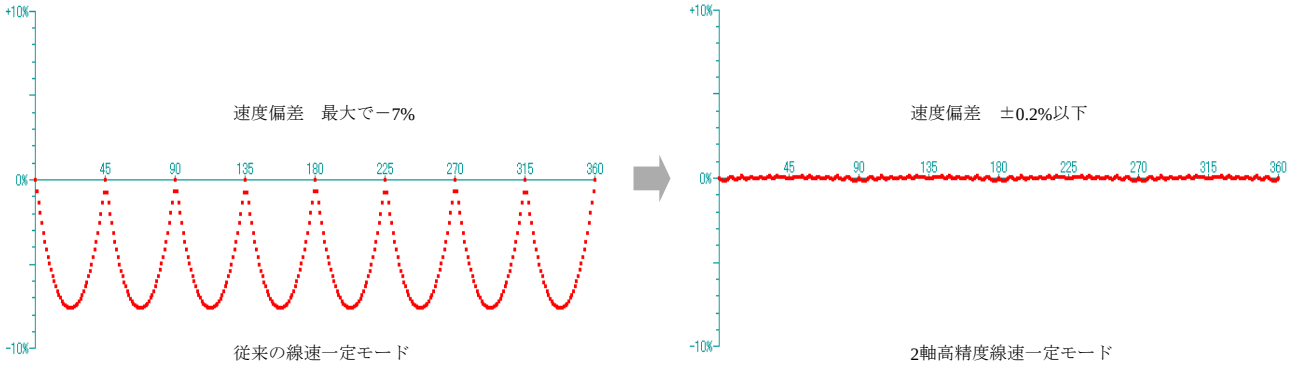


図 1.1-6 線速一定モードの速度偏差

■ 速度レンジフリー

MCX514 は、速度倍率(レンジ)設定をなくした画期的なモーションコントロール IC です。これにより、本 IC の出力速度範囲である 1pps から 8Mpps まで、1pps 単位で全ての速度を設定することができます。

従来の速度倍率を用いた速度設定の場合、

- 低速の細かな速度設定のため、速度倍率を小さく設定 → 高速度のドライブに移行できない
- 高速度ドライブを行うため、速度倍率を大きく設定 → 細かなドライブ速度設定ができない

という制約がありました。
速度レンジフリーの MCX514 はこのような不便さを解消し、ドライブ中に 1pps、2pps のような低速から 1Mpps のような高速パルスへダイレクトに速度変更をさせることが可能です。

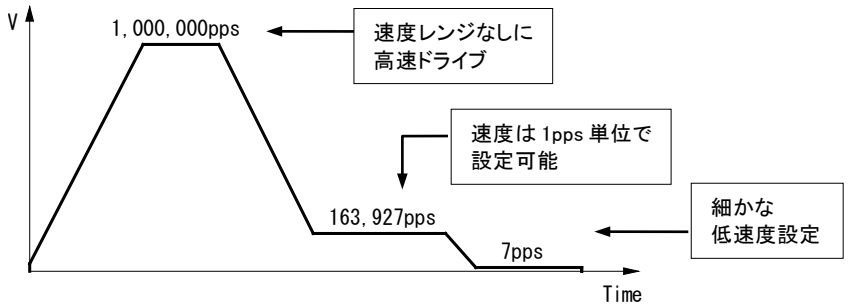


図 1.1-7 速度レンジフリー

■ 簡単で高精度な速度設定

速度倍率の設定がないため、出力するドライブパルス速度をそのまま速度パラメータとして設定可能です。(CLK=16MHz 標準時)

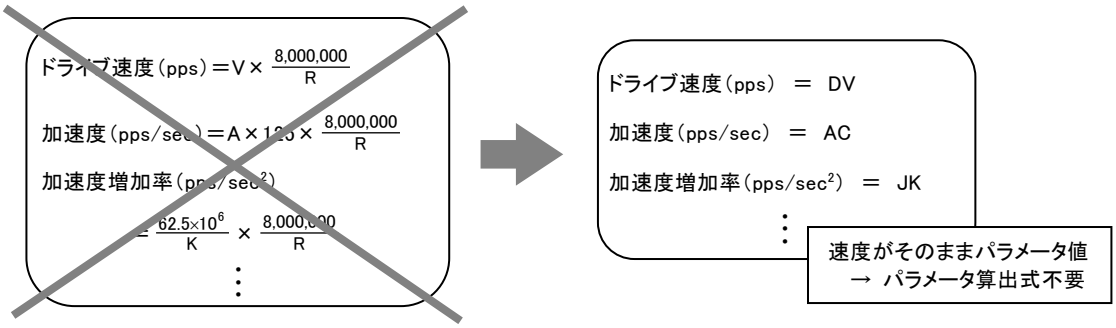


図 1.1-8 速度パラメータ設定

1pps から 8Mpps までの範囲においては、設定されたドライブ速度を精度よく出力します。出力されるドライブパルスの速度精度は、設定値に対して±0.1%以下です。これは、入力 CLK の周波数誤差がないものとした場合です。実際には、入力される CLK に周波数誤差がありますので、これに依存することになります。

■ 豊富な加減速ドライブモード

◆ 加減速ドライブ種類

加減速ドライブは、

定速ドライブ　　直線加減速ドライブ（対称/非対称）　　S字加減速ドライブ（対称/非対称）

を行わせることができます。

◆ 自動減速開始

直線加減速（対称/非対称）、S字加減速ドライブ（対称）の位置ドライブにおいて、減速時は減速開始点をICが算出し、自動的に減速を開始します。（非対称S字加減速ドライブは非対応）

◆ S字加減速カーブ

S字加減速は加速度および減速度を一次直線で増加／減少する方式をとっていますので、速度カーブは2次の放物線加速／減速となります。また、独自の方法によりS字加減速中の三角波形も防止しています。

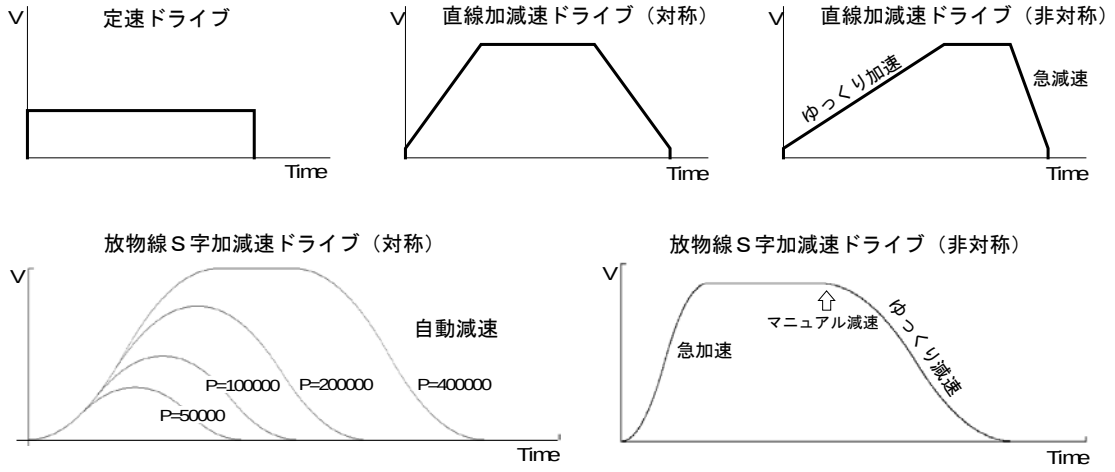


図 1.1-9 加減速ドライブモード

■ ポジション管理機能

ドライブパルス出力をIC内部でカウントする論理位置カウンタと、外部エンコーダからのパルスをカウントする実位置カウンタ、合計2個の 32 ビットポジションカウンタを備えています。
データ読み出し命令により、いつでも現在位置を読み出すことが可能です。
同期動作と組み合わせることで、指定位置通過でドライブ速度変更や他軸のドライブ開始／停止など、位置情報を起動要因にした動作を行うことができます。

■ ソフトリミット機能

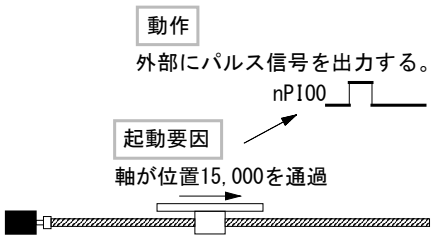
ドライブ中に、位置カウンタが指定範囲を越えるとドライブを停止させるソフトリミット機能を持っています。ソフトリミットを越えた際のドライブ停止方式は減速停止、または即停止から選択可能です。

■ 多彩な同期動作

同期動作は、ある指定の起動要因が発生したら、ある指定の動作を連携して行なう機能です。CPU が介在することなく行うことができるので、指定の動作を高速に精度よく行うことができます。

同期動作は各軸に 4 セット設定可能です。同期動作 1 セットは、指定の 1 つの起動要因と指定の 1 つの動作で構成されます。起動要因として、指定位置通過、ドライブ開始・終了、外部からの入力信号の立ち上がり・立ち下がり、内蔵タイマーのタイムアップなど 15 種類が用意されています。また、動作として、ドライブ開始・停止、現在位置カウンタ値の多目的レジスタへのセーブ、ドライブ速度書き込み、など 28 種類が用意されています。
ある軸の 1 セットの起動要因発生時、同じ軸の他の 3 セット動作、および他軸の 1 セット動作、合計 7 セットの動作を同時に起動させることもできます。

同期動作を複数セット利用することで、より多彩なアプリケーションが実現できます。



- ドライブ中に指定の位置を通過時、外部信号を出力。
- ドライブ中に外部信号入力時、現在位置を所定のレジスタにセーブ。
- ドライブ中に指定位置から外部にスプリットパルスをN個出力。

など

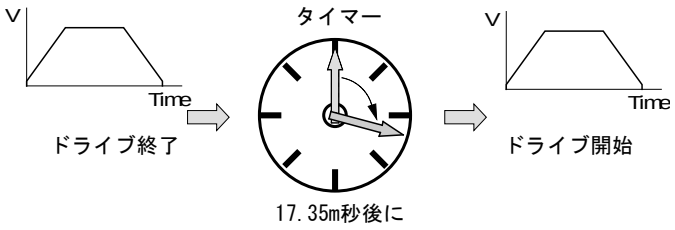
図 1.1-10 同期動作

■ 4個の多目的レジスタ

各軸に 32 ビット長の多目的レジスタを4個装備しています。
多目的レジスタは、現在位置や速度、タイマーとの大小比較を行い、大小関係をステータスで読み取ったり、信号として出力することができるほか、大小関係の変化で同期動作を起動したり、割り込みを発生させることができます。
また同期動作と組み合わせて、ドライブ途中の現在位置や現在速度などの値を多目的レジスタにセーブしたり、あらかじめ多目的レジスタに保存しておいた値を出力パルス数やドライブ速度にロードすることができます。

■ タイマー機能

各軸にタイマーを内蔵しています。1 ～ 2,147,483,647 μ sec の範囲を、1 μ sec 単位で設定します(CLK=16MHz 時)。
同期動作と組み合わせて、下記のような様々な動作を精度よく行うことができます。



- ドライブ終了後、指定時間後にドライブを開始。
- 外部信号入力後、指定時間後にドライブを開始。
- 連続パルスドライブを指定時間後に停止。
- 通過位置Aから通過位置Bまでの移動時間を測定。

など

図 1.1-11 タイマー機能

■ スプリットパルス出力

ドライブ中にスプリットパルスを出力させる機能で各軸が持っています。軸移動と同期を取りながら決められた間隔で種々の動作を行わせるときに使用する機能です。スプリット長、パルス幅、パルス数を設定することができます。同期動作と組み合わせると、指定の位置からスプリットパルスを開始／停止や、外部信号入力でスプリット長やパルス幅の変更などを行うことができます。補間ドライブ時に任意の軸に対応してスプリットパルスを出力させることができます。

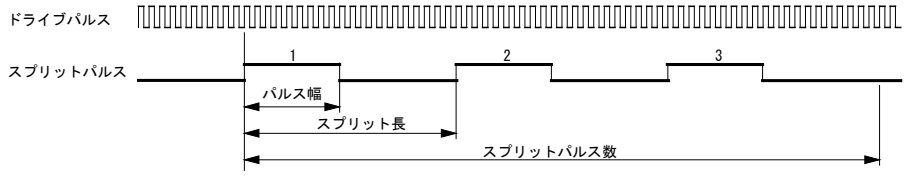


図 1.1-12 スプリットパルス出力

■ 自動原点出し機能

本 IC は、CPU の介在なしに、高速原点サーチ → 低速原点追込み → エンコーダZ相サーチ → オフセット移動などの一連の原点出しシーケンスを自動的に実行する機能を持っています。サーボモータへの偏差カウンタクリア出力も可能です。各ステップ間に停止時間を設けるステップ間タイマーが利用可能です。
回転軸の原点出しにも適した動作も用意されています。

■ サーボモータ用各種信号

2相エンコーダ信号、インポジション、アラームなどのサーボモータドライバ信号を入力できます。また、偏差カウンタクリアのための出力信号も用意されています。

■ 割り込み発生機能

本 IC は2つの割り込み信号(INT0N,INT1N)を持っています。
INT0N 信号では、加減速ドライブ中の定速開始時、定速終了時、ドライブ終了時、位置カウンタと多目的レジスタの大小関係が変化したときなど、様々な要因で割り込みを発生させることができます。
INT1N信号は連続補間ドライブ時にCPUに対し次のセグメントデータ転送を要求するための専用の割り込み信号です。

■ 外部信号によるドライブ操作

外部信号によって、相対位置ドライブ、連続パルスドライブ、手動パルサードライブを行うことができます。この機能により、マニュアルのジョグ送りなどにおいても、上位 CPU のタスクを軽減し、スムーズに動作させることができます。

■ 入力信号フィルタ内蔵

IC内部に、各入力信号の入力段に積分型のフィルタを装備しています。いくつかの入力信号ごとに、フィルタ機能を有効にするか、信号をスルーで通すかを設定できます。また、フィルタの時定数は、16 種類(500nsec～16msec)の中から選択することができます。

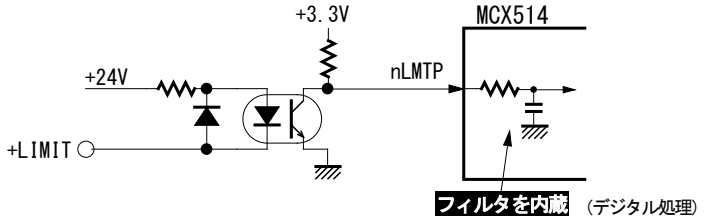


図 1.1-13 入力信号フィルタ内蔵

■ リアルタイムモニタ機能

ドライブ中に現在の論理位置、実位置、ドライブ速度、加速度、加減速状態(加速中、定速中、減速中、加速度増加中、加速度一定、加速度減少中)、タイマーなどをリアルタイムで読み出すことが可能です。

■ CPU インターフェイス

本 IC は上位 CPU との接続インターフェイスとして、従来の 8 ビット/16 ビットデータバスの他に、I²C シリアルインターフェイスバスを備えています。
I²C シリアルインターフェイスバスは、必要なバスラインはシリアル・データライン(SDA) とシリアル・クロックライン(SCL)の 2 本のみなので、端子数の少ない PIC™ マイコンなどを上位 CPU として使用することができます。I²C バスは、同一バス上に複数の MCX514 や EEPROM のような I²C バスインターフェイスを持つデバイスを連結して接続することができます。

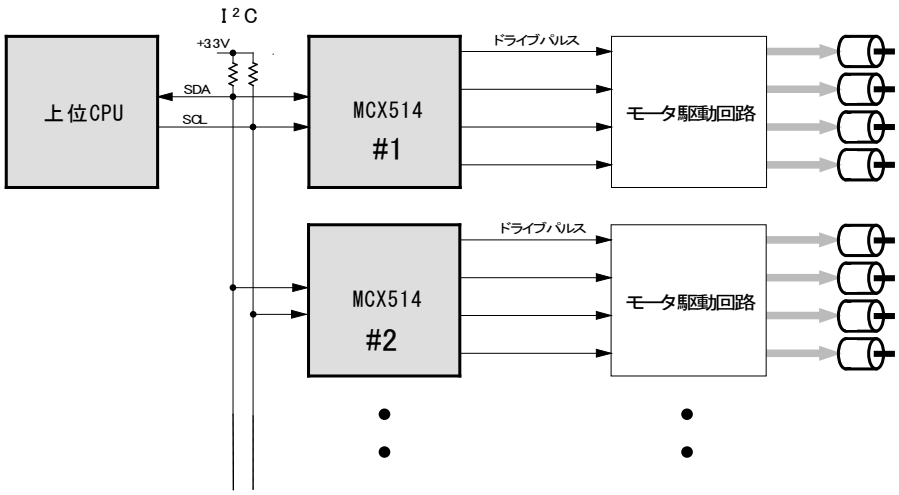


図 1.1-14 I²C シリアルインターフェイスバスの例

1.2 機能ブロック図

図1.2-1に、本ICの機能ブロック図を示します。全く同機能を持つX,Y,Z,Uの4軸の制御部と、補間演算を行う回路ブロックから構成されています。補間ドライブでは、主軸(AX1)に指定された軸の基本パルス発振のタイミングで補間演算が行われます。定速ドライブでも加減速ドライブでも行うことができます。図1.2-2は、各軸の軸制御部の機能ブロック図を示しています。

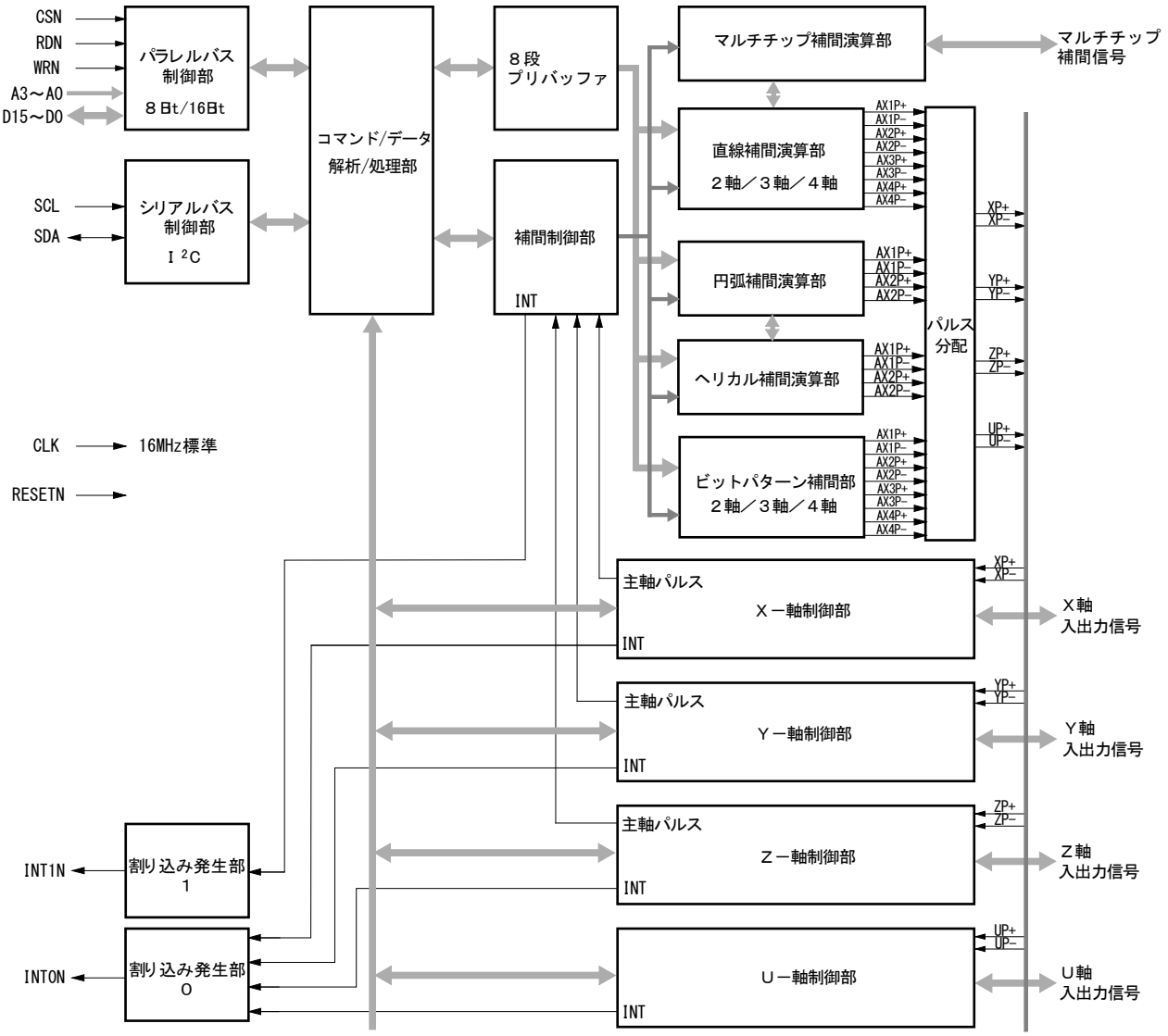
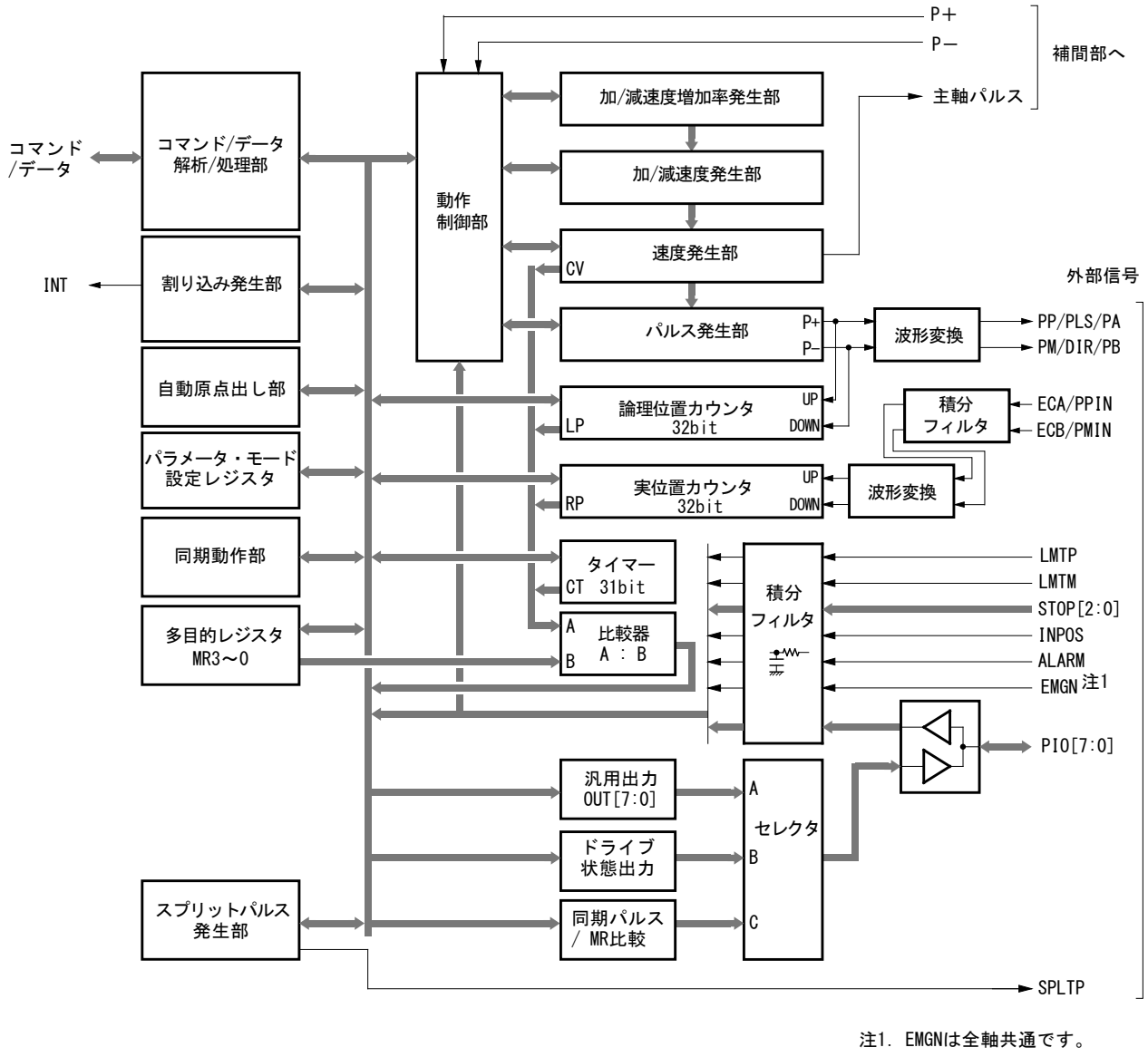


図 1.2-1 MCX514 全体機能ブロック図



注1. EMGNは全軸共通です。

図 1.2-2 X、Y、Z、Uー軸制御部内のブロック図(1軸分のみ記載)

1.3 仕様一覧

（CLK＝16MHz 時）

項目	小項目	内容	補足
制御軸数		4 軸	
CPU パラレルバス接続		16 ビットまたは 8 ビットバス選択可能	
CPU シリアルバス接続		I^2C シリアルインターフェイスバス	
補間機能	補間命令	2 軸/3 軸/4 軸直線補間、CW/CCW 円弧補間、 2 軸/3 軸/4 軸ビットパターン補間、CW/CCW ヘリカル補間	
	補間範囲	各軸 −2,147,483,646 ～ 2,147,483,646 drive pulse	
	補間速度	1 pps ～ 8,000,000 pps	*11
	補間位置精度	±0.5LSB 以下（直線補間） ±1LSB 以下（円弧補間）	
	その他補間に関する機能	・任意軸選択可能 ・短軸パルス均一 ・線速一定(2 軸/3 軸簡単モード、2 軸高精度モード選択可能） ・連続補間 ・8 段プリレジスタによるデータバッファリング ・補間ステップ送り ・マルチチップ多軸直線補間	
ドライブパルス出力	ドライブ速度範囲	1 pps ～ 8,000,000 pps （CLK＝20MHz 時：最高 10,000,000pps）	
	初速度範囲	1 pps ～ 8,000,000 pps	
	出力速度精度	±0.1%以下（設定値に対して）	
	加速度範囲	1 pps/sec ～ 536,870,911pps/sec	
	加速度増加・減少率範囲	1 pps/sec^2 ～ 1,073,741,823 pps/sec^2	*1
	加減速カーブ	定速、対称／非対称直線加減速、対称／非対称S字加減速	
	ドライブパルス範囲	・相対位置ドライブ : −2,147,483,646 ～ 2,147,483,646 drive pulse ・絶対位置ドライブ : −2,147,483,646 ～ 2,147,483,646 drive pulse	*2
	位置ドライブ減速停止モード	自動減速停止／マニュアル減速停止	*3
	オーバライド	ドライブ中の出力パルス数、ドライブ速度の変更可能	*4
	ドライブ命令の種類	相対位置、絶対位置、＋方向連続、－方向連続	
	三角防止機能	直線加減速、S字加減速ともに有り	
	ドライブパルス出力方式	独立2パルス、1パルス・方向、2 相 4 逓倍、2 相 2 逓倍 選択可能	
	ドライブパルス出力論理	正論理／負論理出力 選択可能	
	ドライブパルス出力端子	端子入れ替え可能	
エンコーダ入力	入力パルス入力方式	2 相 4 逓倍／2 相 2 逓倍／2 相 1 逓倍／アップダウンパルス 選択可能	
	入力パルス端子	端子入れ替え可能	
位置カウンタ	論理位置カウンタ	カウント範囲：−2,147,483,648 ～ +2,147,483,647 drive pulse	*5
	実位置カウンタ	カウント範囲：−2,147,483,648 ～ +2,147,483,647 pulse	*5
	可変リング	各カウンタのカウント最大値を設定可能	
ソフトリミット	設定範囲	−2,147,483,648 ～ +2,147,483,647 pulse	
	停止モード	減速停止／即停止 選択可能	
多目的レジスタ	ビット長・個数	32 ビット長・4 個／軸	
	用途	位置・速度・タイマー値比較、位置・速度などのロード、 ドライブ中の現在位置・速度・タイマー値などをセーブ	
タイマー	機能数	各軸 1 個	
	設定範囲	1 ～ 2,147,483,647 μsec	
スプリットパルス	信号点数	各軸1点	
	スプリット長	2 ～ 65,535 drive pulse	*6
	スプリットパルス幅	1 ～ 65,534 drive pulse	
	スプリットパルス数	1 ～ 65,535、または無限	
自動原点出し	シーケンス	STEP1 高速原点サーチ → STEP2 低速原点追込み → STEP3 エンコーダZ相サーチ → STEP4 オフセット移動 ・各ステップの有効／無効、検出信号、検出方向の選択可能。	
	偏差カウンタクリア出力	クリアパルス幅 10 μ～20msec 内から選択、論理レベル選択可能	
	ステップ間タイマー	1msec ～ 1,000msec 内から選択	

同期動作	セット数	各軸 4 セット	*7
	起動要因	・多目的レジスタ比較変化時 ・比較対象：論理／実位置カウンタ値、現在速度値、現在タイマー値 ・比較条件：≧，＞，＝，＜ ・タイマーのタイムアップ時 ・ドライブ開始／終了時、加減速ドライブの定速域開始／終了時 ・スプリット動作開始／終了時、スプリットパルス出力時 ・nPIOm 信号↑／↓、nPIOm+4 信号 Low かつ nPIOm 信号↑、 nPIOm+4 信号 Hi かつ nPIOm 信号↑、nPIOm+4 信号 Low かつ nPIOm 信号↓、nPIOm+4 信号 Hi かつ nPIOm 信号↓ （m:0,1,2,3） ・起動命令	
	動作	・値のロード（MRm→設定値）：ドライブ速度、移動パルス数（終点）、スプリット長、スプリットパルス幅、論理位置カウンタ値、実位置カウンタ値、初速度、加速度 ・値のセーブ（MRm←現在値）：論理位置カウンタ値、実位置カウンタ値、現在タイマー値、現在ドライブ速度、現在加減速度 ・外部への同期パルス出力 ・相対／絶対位置ドライブ起動、＋／－方向連続ドライブ起動、MRm に設定された位置データでの相対／絶対位置ドライブ起動 ・ドライブ減速停止／即停止、速度増加／減少、タイマー始動／停止、スプリットパルス動作開始／停止	
	他 SYNC 起動	他の 3 セットの動作起動設定可能	
	他軸 SYNC0 起動	他軸の SYNC0 の動作起動設定可能	
	繰返し	同期動作の単一／繰り返し設定可能	
割り込み	信号数	INT0N、INT1N の 2 本	
	割り込み発生要因	・多目的レジスタ比較変化時 ・比較対象：論理／実位置カウンタ値、現在速度値、現在タイマー値 ・比較条件：≧，＞，＝，＜ ・ドライブ開始／終了時、加減速ドライブの定速域開始／終了時 ・自動原点出し終了時、タイマー終了時 ・スプリットパルス出力時、スプリット動作終了時 ・同期動作 0／1／2／3 起動時 ・8 段プリバッファレジスタ状態変化時(連続補間ドライブ)	
	許可／禁止	各割り込み要因の有効／無効 選択可能	
外部信号による ドライブ操作		・EXPP, EXPM 信号による相対位置ドライブ、連続ドライブ ・手動パルサー（エンコーダ入力：2 相 1 逓倍） ・EXPLSN 信号による補間ステップ送り	*8
外部停止信号	信号点数	3 点 （STOP0～2）／軸	
	有効／無効	停止信号機能の有効／無効 選択可能	*9
	論理レベル	Low アクティブ／Hi アクティブ 選択可能	
	停止モード	アクティブ時、ドライブ減速停止 （初速度以下でドライブ時は即停止）	
サーボモータ用 入出力信号	信号種類	ALARM（アラーム）／軸、INPOS（位置決め完了）／軸、DCC（偏差カウンタクリア）／軸	
	有効／無効	信号の有効／無効 選択可能	
	論理レベル	Low アクティブ／Hi アクティブ 選択可能	
汎用入/出力信号	信号点数	8 点／軸 ・同期入力、外部ドライブ入力信号端子と共用 ・同期動作出力、多目的レジスタ比較出力、ドライブ状態出力信号端子と共用	
ドライブ状態出力 信号	信号の種類	・ドライブ中、エラー中、加速／定速／減速中、加速度増加／一定／減少中 ・ドライブ状態はステータスレジスタでも読み出し可能	*10
オーバランリミット 信号	信号点数	各軸 2 点 （＋方向、－方向 各 1 点）	
	有効／無効	リミット機能の有効／無効 選択可能	*9
	論理レベル	Low アクティブ／Hi アクティブ 選択可能	
	停止モード	アクティブ時のドライブ即停止／減速停止 選択可能	
	入力端子	端子入れ替え可能	

緊急停止信号		全軸で EMGN 1 点　Low レベルでドライブパルス出力を停止 （論理レベル設定不可）	
積分型フィルタ内蔵	入力信号フィルタ	各信号の入力段に積分フィルタを装備	
	時定数	16 種類から選択可能 （500n, 1μ 2μ 4μ 8μ 16μ 32μ 64μ 128μ 256μ 512μ 1m, 2 m, 4 m, 8 m, 16 m[sec]）	
	有効／無効	フィルタ機能の有効／無効 選択可能	
電気的特性	動作温度範囲	−40℃～+85℃	
	動作電源電圧	+3.3V ±10%	
	消費電流	150mA（平均）, 204mA（最大） CLK=16MHz 時	
	入力クロック周波数	16MHz（標準） 20MHz（最大）	
	入力信号レベル	TTL レベル（5V トレラント）	
	出力信号レベル	3.3V CMOS レベル（5V 系には TTL のみ接続可能）	
パッケージ		・144 ピン・プラスチック QFP 0.5mm ピンピッチ RoHS 指令対応品 ・パッケージサイズ 20×20×1.4 mm	

＜補足＞

記号	
*1	S字加減速ドライブで使用するパラメータです。
*2	指定されたパルス数を出力するドライブ時の設定できるパルス範囲です。連続ドライブ時は無限パルスを出力します。
*3	自動減速停止は、指定の移動パルス量から減速開始位置を IC 内で計算し自動的に減速停止させます。マニュアル減速停止は、上位から減速開始位置を設定して減速停止させます。本 IC は非対称S字加減速を除き自動減速停止させることができます。
*4	ドライブ開始後の出力パルス数の変更は、同一方向への相対位置ドライブのみ可能です。 連続補間ドライブの途中でドライブ速度を変更することはできません。
*5	論理位置カウンタは出力するドライブパルスをカウントします。実位置カウンタはエンコーダ入力パルスをカウントします。
*6	ドライブ中にドライブパルスと同期して、指定のパルス間隔でスプリットパルスを出力します。
*7	同期動作 1 セットは、指定の 1 つの起動要因と指定の 1 つの動作から成ります。
*8	外部信号入力端子は、汎用入出力端子と共用です。
*9	機能を使用しないとき、汎用入力として使用可能です。
*10	ドライブ状態出力端子は、汎用入出力端子と共用です。
*11	ビットパターン補間は 4Mpps 以下、ヘリカル補間は 2Mpps 以下、連続補間は 4Mpps 以下、マルチチップ補間は 4Mpps 以下です。

2. 機能説明

2.1 定量パルスドライブと連続パルスドライブ

ドライブパルス出力には、予め出力パルス数を定めて行う定量パルスドライブと、停止命令が発行されるまで、または停止信号が入力されるまでパルスを出力し続ける連続パルスドライブがあります。定量パルスドライブには、相対位置ドライブ，絶対位置ドライブ，反相対位置ドライブがあります。連続パルスドライブには、＋方向連続パルスドライブ，－方向連続パルスドライブがあります。

- 定量パルスドライブ
  - 相対位置ドライブ
  - 絶対位置ドライブ
  - 反相対位置ドライブ
- 連続パルスドライブ
  - ＋方向連続パルスドライブ
  - －方向連続パルスドライブ

2.1.1 相対位置ドライブ

相対位置ドライブは、現在位置から移動パルス数を指定してドライブを行います。現在位置に対して＋方向へ移動させるときは、移動パルス数に正のパルス値をセットし、－方向へ移動させる時は負のパルス値をセットします。

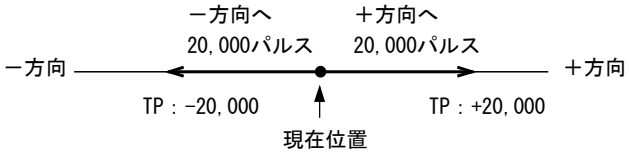


図 2.1-1 相対位置ドライブの移動パルス数(TP)設定 例

相対位置ドライブは、定速または加減速ドライブを行います。加速度と減速度が等しい加減速での相対位置ドライブの動作は、図 2.1-2 で示すように、移動パルスの残りが加速時に消費されたパルス数より小さくなると自動減速を開始し、指定の移動パルス数を出力し終えるとドライブを終了します。

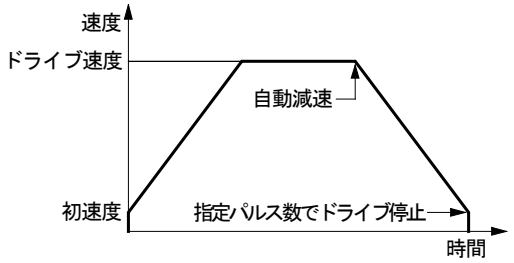


図 2.1-2 相対位置ドライブの自動減速と停止

相対位置ドライブの命令コードは 50h です。直線加減速で相対位置ドライブを行うには、次のパラメータを設定する必要があります。

表 2.1-1 設定パラメータ：相対位置ドライブ

パラメータ名	記号	コメント
加/減速度	AC／DC	加速と減速が等しい時は減速度の設定は不要
初速度	SV	
ドライブ速度	DV	
移動パルス数／終点	TP	＋方向へのドライブは＋のパルス数、 －方向へのドライブは－のパルス数を指定

2.1.2 絶対位置ドライブ

絶対位置ドライブは、原点(論理位置カウンタ値:0)を基準にした移動先の位置を指定してドライブを行います。移動先の位置を、現在位置にかかわらず絶対座標で指定することができます。指定した移動先位置と現在位置の差から、ドライブ方向と出力パルス数を IC 内部で算出しドライブを行います。
絶対位置ドライブでは移動先を移動空間内の座標で指定するため、あらかじめ原点出しを行い、論理位置カウンタを確定してからドライブを行う必要があります。

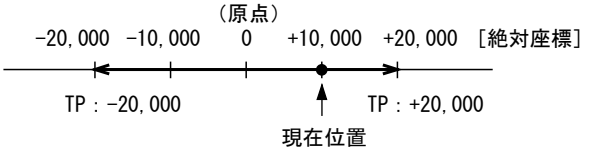


図 2.1-3 絶対位置ドライブの終点(TP)指定 例

絶対位置ドライブは、相対位置ドライブと同様、定速または加減速ドライブを行います。
絶対位置ドライブの命令コードは 54h です。直線加減速で絶対位置ドライブを行うには、次のパラメータを設定する必要があります。

表 2.1-2 設定パラメータ : 絶対位置ドライブ

パラメータ名	記号	コメント
加/減速度	AC／DC	加速と減速が等しい時は減速度の設定は不要
初速度	SV	
ドライブ速度	DV	
移動パルス数／終点	TP	移動先の絶対座標を指定

2.1.3 反相対位置ドライブ

反相対位置ドライブは、現在位置を基準にした移動先位置への方向と移動パルス数を指定してドライブを行います。相対位置ドライブとは異なり、移動パルス数(TP)に設定したパルス数の符号と反対方向へドライブします。
移動パルス数にある定まった正のパルス値をあらかじめ設定し、ドライブ命令でドライブする方向を決定したいときに使用します。
移動パルス数に負のパルス値が設定されている場合、反相対位置ドライブは＋方向にドライブします。

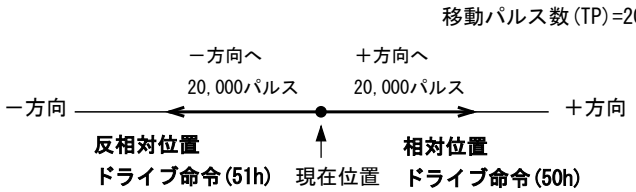


図 2.1-4 相対／反相対位置ドライブ命令によるドライブ方向の決定

反相対位置ドライブは、移動パルス数に設定したパルス値の符号と逆方向へドライブする以外の動作は相対位置ドライブと同様です。反相対位置ドライブの命令コードは 51h です。

■ ドライブ途中の移動パルス数の変更(オーバライド)

相対位置ドライブおよび反相対位置ドライブの途中で、移動パルス数(TP)を変更することができます。ただし、移動パルス数の変更前と変更後でドライブの方向が同じである必要があります。ドライブの方向が変わるような値に、移動パルス数を変更することはできません。

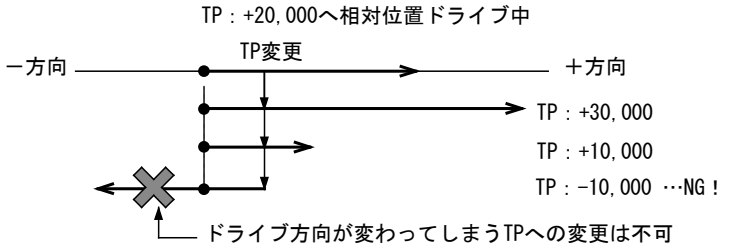


図 2.1-5 相対位置ドライブの移動パルス数（TP）オーバライド

加減速でドライブ中、出力パルスの残りが加速時のパルスより少なくなり、減速に入っているときに移動パルス数(TP)が変更された場合は、再び加速を始めます(図 2.1-7)。また、変更した移動パルス数(TP)による出力パルス数がすでに出し終えたパルス数より小さい場合は、即停止します(図 2.1-8)。
S字加減速では、減速時に移動パルス数(TP)が変更されると正しいS字カーブを描くことができませんので、ご注意ください。

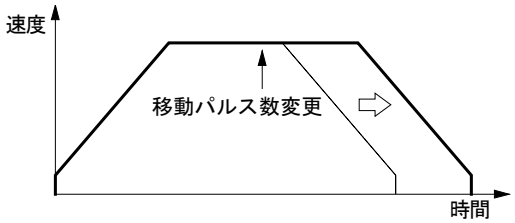


図 2.1-6 ドライブ中の移動パルス数変更

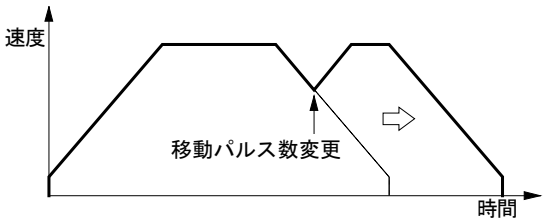


図 2.1-7 減速時の移動パルス数変更

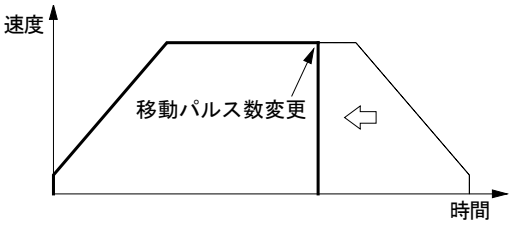


図 2.1-8 出力されたパルスより少ない移動パルス数に変更

【注意】絶対位置ドライブはドライブ途中で移動パルス数(TP)を変更することはできません。

■ 加減速定量パルスドライブにおけるマニュアル減速

加減速ドライブの定量パルスドライブ（相対位置ドライブ、絶対位置ドライブ、反相対位置ドライブ）では、通常 図 2.1-2 に示すように、ICが算出した減速点で自動減速開始しますが、この減速点をマニュアルで指定することもできます。
下記のような場合には、自動減速点がはずれてきたり、まったく算出できなくなりますので、マニュアルで減速点を指定しなければなりません。

- **直線加減速の定量パルスドライブにおいて、ドライブ途中に速度変更をたびたび行う。**
- **非対称台形加減速、およびS字加減速の定量パルスドライブにおいて、ドライブ途中に速度変更を行う。**
- **S字加減速の定量パルスドライブにおいて、加速度と減速度、加速度増加率と減速度増加率を個別設定する。（非対称S字加減速）**
- **円弧補間、ビットパターン補間、連続補間を加減速で行う。**

マニュアル減速モードにするには、WR3 レジスタの D0 ビットを1にし、マニュアル減速点設定命令(07h)によって減速点をセットします。その他の操作は、通常の定量パルスドライブと同様です。

■ 加減速定量パルスドライブにおける加速カウンタオフセット

加減速の定量パルスドライブの動作では、加速時に、加速で消費されるパルスを加速カウンタでカウントします。加減速度が等しい加減速ドライブにおいては、設定されている出力パルス数の残りが加速カウンタの値より少なくなると減速を開始し、減速中に加速中と同じパルス数を出力するようにしています。

加速カウンタオフセットは、この加速カウンタに指定のオフセット値を加算します。図 2.1-9 に示すように、オフセット値を正の値で大きくするほど、自動減速ポイントが手前に移動してきますので、減速終了時の初速度での引き摺りが長くなります。また、オフセット値を負の値でセットすると初速度まで落ちきらずに尻切れで停止する傾向になります。

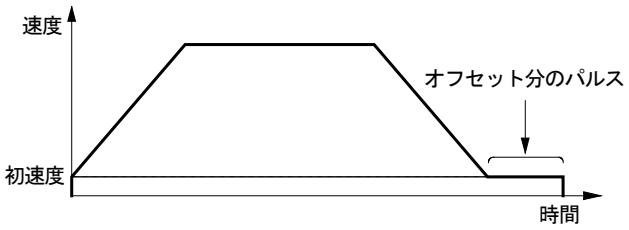


図 2.1-9 加速カウンタオフセット

加速カウンタオフセットはリセット時、0 にセットされます。通常の直線加減速ドライブを行う場合には、このパラメータを再設定する必要はほとんどありません。非対称台形加減速やS字加減速の定量パルスドライブで、初速度を低く設定したためにドライブ終了時の引き摺りパルスや尻切れが問題になるときに、加速カウンタオフセットを適当な値にセットして補正します。

2.1.4 連続パルスドライブ

連続パルスドライブは、上位からの停止命令、または外部からの停止信号がアクティブになるまで、連続してドライブパルスを出し続けます。原点サーチ、スキャニングジョグ送り、あるいは速度制御でモータを回転させるときなどに使用します。
停止命令には、減速停止命令と、即停止命令があります。また、外部からの減速停止（初速度以下でドライブ時は即停止）信号は nSTOP0～nSTOP2 の 3 点が用意されています。各々の信号は、有効／無効、アクティブレベルをモード設定することができます。

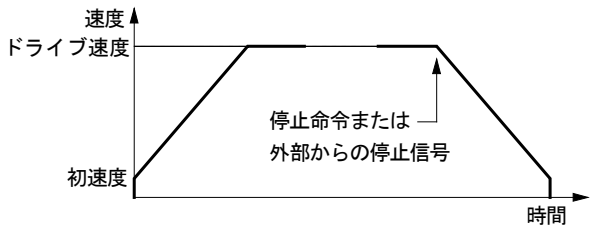


図 2.1-10 連続パルスドライブ

連続パルスドライブ命令は、＋方向連続パルスドライブ命令(52h)と、－方向連続パルスドライブ命令(53h)が用意されています。
連続パルスドライブを加減速で行うには、移動パルス数(TP)以外は、定量パルスドライブと同様のパラメータを設定する必要があります。

表 2.1-3 設定パラメータ：連続パルスドライブ

パラメータ名	記号	コメント
加/減速度	AC／DC	加速と減速が等しい時は減速度の設定は不要
初速度	SV	
ドライブ速度	DV	

■ ドライブ途中のドライブ速度の変更(オーバライド)

連続パルスドライブでは、ドライブ途中でドライブ速度を自由に変更することができます。ドライブ速度変更方法は、ドライブ速度パラメータ(DV)を変更する方法と、速度増加，減少命令による方法が用意されています。
ただし、S字加減速ドライブにおいては、加減速中に速度変更をかけても無効になります。

定量パルスドライブの対称直線加減速、および定速ドライブにおいても、ドライブ途中でドライブ速度を変更することができます。
ただし、直線加減速の定量パルスドライブにおいて、ドライブ速度を変更すると、若干の尻切れが発生する場合がありますので、低い初速度設定で使用する場合にはご注意ください。
なお、定量パルスドライブ(自動減速モード時)の非対称直線加減速、およびS字加減速では、ドライブ途中でドライブ速度を変更することはできません。

＜ドライブ速度パラメータ変更による速度変更＞
ドライブ速度設定命令(05h)によりドライブ速度パラメータ(DV)を変更すると、直ちに設定が反映され、加減速ドライブであれば指定のドライブ速度まで加減速します。

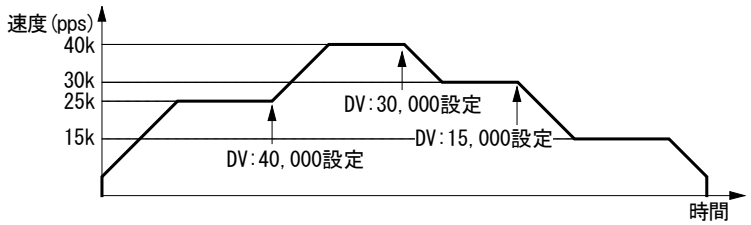


図 2.1-11 ドライブ途中のドライブ速度変更 例

＜速度増加，減少命令による速度変更＞
あらかじめ速度増減値パラメータ(IV)に増減する速度を設定します。ドライブ途中の速度増加命令(70h)、速度減少命令(71h)発行で直ちに設定が反映され、加減速ドライブであれば現在速度から速度増減値パラメータ分、加減速します。

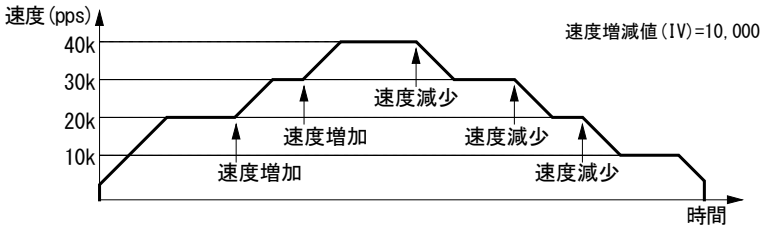


図 2.1-12 速度増加，減少命令による速度変更 例

【注意】ドライブ途中でドライブ速度を変更する場合には、三角波形防止機能を無効(WR3／D13：1)にしてください。

■ 連続パルスドライブによる原点検出動作

原点近傍信号、原点信号、エンコーダZ相信号などを nSTOP0～nSTOP2 に割り当てます。(エンコーダZ相信号は nSTOP2 に割り当ててください。)WR2 レジスタで各信号の有効／無効、論理レベルを設定します。高速サーチの場合は、加減速で連続パルスドライブを行います。有効に設定した信号がアクティブレベルになると減速停止します。低速サーチの場合は、定速で連続パルスドライブを行います。有効に設定した信号がアクティブレベルになると即停止します。
本 IC には自動原点出し機能も用意されています。自動原点出し機能については 2.5 節を参照してください。

2.2 加減速

ドライブパルス出力の速度カーブは、加減速を行わない定速ドライブ、目標速度へ直線的に加減速を行う直線加減速ドライブ、目標速度へ滑らかなカーブで加減速を行うS字加減速ドライブがあります。
加減速ドライブは、加速度と減速度が等しい対称加減速、および加速度と減速度を個別に設定する非対称加減速がそれぞれ用意されています。

- 定速ドライブ
- 加減速ドライブ
  - 直線加減速ドライブ
    - 直線加減速(対称)
    - 非対称直線加減速
  - S字加減速ドライブ
    - S字加減速(対称)
    - 非対称S字加減速

2.2.1 定速

定速ドライブは加減速を行わず、常に一定の速度でドライブパルスを出力します。定速ドライブを行うには、ドライブ速度は初速度以下の値(つまり初速度がドライブ速度以上の値)に設定します。定速ドライブは、初速度以下であるドライブ速度で加減速せずドライブを行います。停止動作は即停止となります。
原点サーチやエンコーダのZ相サーチなど信号を検出したら即停止させたい時は、加減速ドライブを行わず始めから低スピードの定速ドライブを行います。

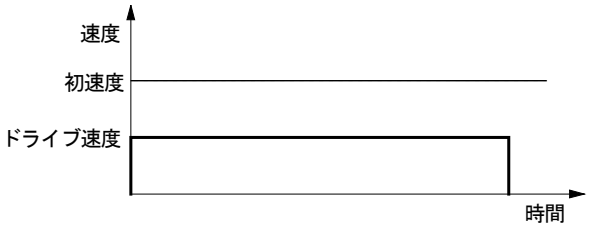


図 2.2-1 定速ドライブ

定速ドライブを行うには、次のパラメータを設定する必要があります。

表 2.2-1 設定パラメータ：定速ドライブ

パラメータ名	記号	コメント
初速度	SV	ドライブ速度 (DV) 以上の速度値を設定
ドライブ速度	DV	
移動パルス数／終点	TP	連続パルスドライブでは不要

■ パラメータ設定例

ドライブ速度 980pps の定速ドライブを行う例を示します。この例では、移動パルス値 2,450 の相対位置ドライブを行います。

初速度	SV = 980	初速度≧ドライブ速度の値を設定
ドライブ速度	DV = 980	
移動パルス数	TP = 2450	

各パラメータについては 7.2 節を参照してください。

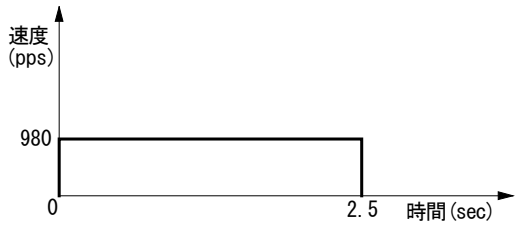


図 2.2-2 定速ドライブ 例

2.2.2 直線加減速(対称)

直線加減速ドライブは、ドライブ開始の初速度からドライブ速度まで、指定の加速度の傾きを持つ一次直線で加速します。直線加減速ドライブは自動減速が可能で、マニュアル減速点の設定が不要です。加速度と減速度が等しい対称直線加減速の定量パルスドライブにおいて、加速中に消費するパルス数をカウントし、出力パルス数の残りが加速中にカウントしたパルス数より少なくなると自動的に減速を開始します。減速は加速度と同じ傾きを持つ一次直線で初速度まで減速を行い、すべての出力パルス数を出し終えると停止します。
加速中に減速停止がかかった場合は、図 2.2-3 のように加速途中から減速を行います。

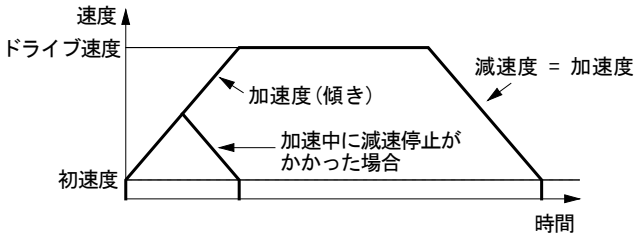


図 2.2-3 直線加減速ドライブ（対称）

対称の直線加減速ドライブを自動減速で行うには、WR3 レジスタの D2～0 ビットを次のように設定します。また、次のパラメータを設定する必要があります。

表 2.2-2 モード設定：直線加減速（対称）

モード設定ビット	記号	設定値	コメント
WR3/D0	MANLD	0	自動減速
WR3/D1	DSNDE	0	減速時に加速度設定値を使用する（対称）
WR3/D2	SACC	0	直線加減速

表 2.2-3 設定パラメータ：直線加減速（対称）

パラメータ名	記号	コメント
加速度	AC	減速時はこの値で減速する
初速度	SV	
ドライブ速度	DV	
移動パルス数／終点	TP	連続パルスドライブでは不要

■ パラメータ設定例

初速度:500pps、ドライブ速度:15,000pps まで 0.3 秒で直線加減速を行うパラメータ設定を以下に示します。

加速度	AC = 48333	(15000-500)/0.3
		= 48333pps/sec
初速度	SV = 500	
ドライブ速度	DV = 15000	

各パラメータについては 7.2 節を参照してください。

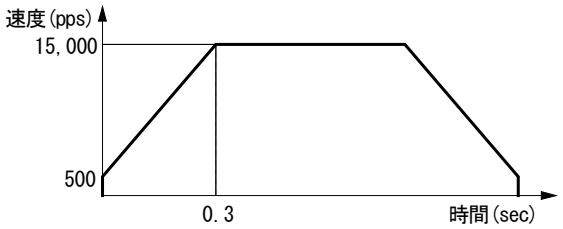


図 2.2-4 直線加減速ドライブ（対称） 例

■ 直線加減速の三角波形防止（定量パルスドライブ）

三角波形防止機能は、直線加減速の定量パルスドライブにおいて、出力パルス数がドライブ速度までの加速で必要とするパルス数に満たない場合でも、三角波形を防止する機能です。三角波形とは、直線加減速ドライブにおいて加速中に減速に転じる速度カーブをいいます。
三角波形防止機能によって、加速中に加速時と減速時に消費するパルス数の合計が定量ドライブの全出力パルス数の 1/2 を越えると加速を停止し、その速度を維持してドライブを行った後、自動減速します。従って定量パルスドライブの出力パルス数が少ない場合でも出力パルス数の 1/2 が定速域となり、三角波形を台形波形にすることができます。

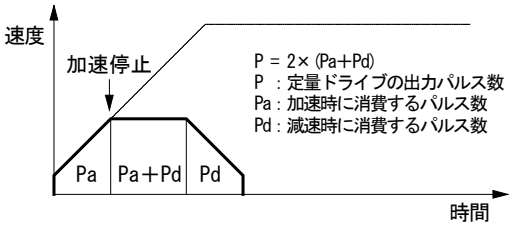


図 2.2-5 直線加減速ドライブの三角波形防止

直線加減速定量パルスドライブの三角波形防止機能は、リセット時から有効になっています。WR3 の D13 ビットを 1 に設定することで、本機能を無効にする事も可能です。

直線加減速ドライブにおいて、加速中に減速停止がかかった場合、三角波形防止にはなりません。図 2.2-3 で示すように減速停止がかかった時点から減速を開始します。

【注意】ドライブ途中でドライブ速度を変更する場合には、三角波形防止機能を無効(WR3／D13：1)にしてください。

2.2.3 非対称直線加減速

さまざまなワークのスタッキング装置などでは、垂直方向に対象物を動かす場合、対象物に対して重力加速度が加わるために上下移動の加速度と減速度を変えたい場合があります。
本 IC は、このように加速度と減速度の異なる非対称直線加減速の定量パルスドライブにおいても、自動減速を行うことができます。あらかじめ計算によってマニュアル減速点を設定しておく必要はありません。図 2.2-6 は、加速度より減速度が大きい例、図 2.2-7 は減速度より加速度が大きい例です。このような非対称の直線加減速においても、定量パルスドライブの出力パルス数と、各速度パラメータ値から自動減速開始点を IC 内部で算出します。

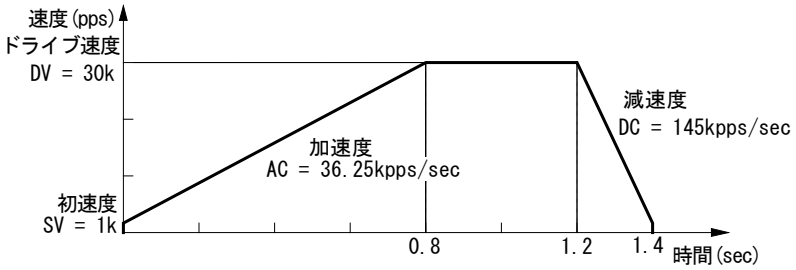


図 2.2-6 非対称直線加減速ドライブ（加速度＜減速度）

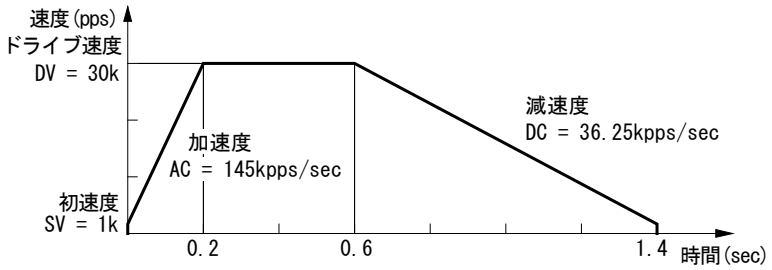


図 2.2-7 非対称直線加減速ドライブ（加速度＞減速度）

非対称直線加減速ドライブを自動減速で行うには、WR3 レジスタの D2～0 ビットを次のように設定します。また、次のパラメータを設定する必要があります。

表 2.2-4 モード設定：非対称直線加減速

モード設定ビット	記号	設定値	コメント
WR3/D0	MANLD	0	自動減速
WR3/D1	DSNDE	1	減速時に減速度設定値を使用する
WR3/D2	SACC	0	直線加減速

表 2.2-5 設定パラメータ：非対称直線加減速

パラメータ名	記号	コメント
加速度	AC	
減速度	DC	
初速度	SV	
ドライブ速度	DV	
移動パルス数／終点	TP	連続パルスドライブでは不要

【注意】

- 非対称直線加減速ドライブにおいて、加速度＞減速度(図 2.2-7)の場合、加速度と減速度の比率に次の条件があります。

$$DC > AC \times \frac{DV}{8 \times 10^6}$$

DC：減速度(pps/sec)
AC：加速度(pps/sec)　　ただし CLK = 16MHz
DV：ドライブ速度(pps)

例えば、ドライブ速度 DV = 100kpps とすると、減速度DC は加速度AC の値の1/80 より大きな値にしなければなりません。1/80 より小さくすることはできません。

- 非対称直線加減速ドライブにおいて、加速度＞減速度(図 2.2-7)の場合、加速度 AC と減速度 DC の比率が大きくなればなるほど引き摺りパルスが多くなります(AC/DC = 10 倍で最大 10 パルス程度)。引き摺りパルスが問題になる場合には、初速度を上げる、加速カウンタオフセットにマイナス値をセットする、等で対処します。

■ パラメータ設定例

前記、図 2.2-6 に示す非対称直線加減速(加速度＜減速度)相対位置ドライブのパラメータ設定は、以下のようになります。

モード設定	WR3←0002h	WR3 レジスタのモード設定
加速度	AC = 36250	(30000-1000)/0.8 = 36250pps/sec
減速度	DC = 145000	(30000-1000)/0.2 = 145000pps/sec
初速度	SV = 1000	
ドライブ速度	DV = 30000	
移動パルス数	TP = 27500	相対位置ドライブ

【注意】非対称直線加減速ドライブにおいても三角波形防止機能は働きますが、ドライブ途中でドライブ速度を変更する場合には、三角波形防止機能を無効(WR3／D13：1)にしてください。

2.2.4 S 字加減速(対称)

S字加減速ドライブは、指定のドライブ速度まで2次曲線(放物線)による滑らかな速度カーブで加減速を行います。
本 IC は、ドライブ速度の加速および減速時において、加速度／減速度を一次直線で増加／減少させることにより、速度のS字カーブを作り出します。
加速と減速が対称なS字加減速ドライブは、図 2.2-8 に示すような動作を行います。

区間a. ドライブが開始されると加速が始まり、加速度は 0 から指定の加速度増加率で直線増加します。このときの速度カーブは、2 次曲線になります。

区間b. 目的のドライブ速度と現在速度との差が、加速度増加中に消費した速度分より少なくなると、加速度は指定の加速度増加率で直線減少します。加速度減少の割合は加速度増加時と同じです。このときの速度カーブは逆向きの放物線になります。

区間c. 速度が指定のドライブ速度に達すると、または加速度が 0 に到達すると、その速度を維持しドライブを行います。
加速と減速が対称なS字加減速の定量パルスドライブでは、出力パルス数の残りが加速で消費したパルス数より小さくなると減速を開始します。(自動減速)

区間d,e. 減速時においても、加速時と同様に、減速度を一次直線で増加／減少させて、速度のS字カーブを生成します。

また、連続パルスドライブ途中でドライブ速度が変更した場合の加速／減速においても、同様の動作を行います。ただし、S字加減速ドライブにおいて、加減速中に速度変更をかけても無効になります。

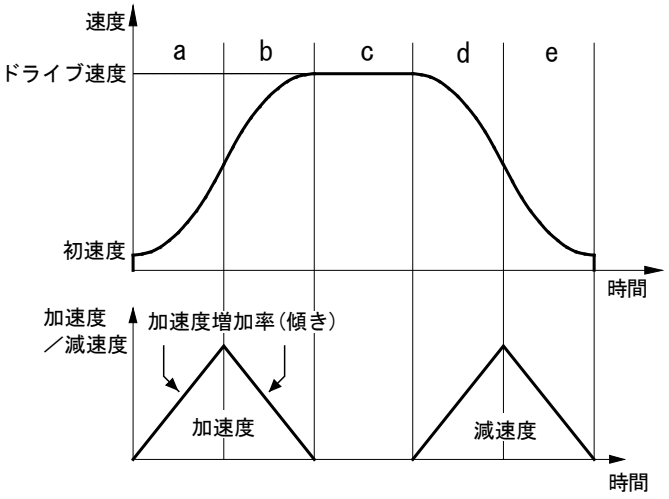


図 2.2-8 S字加減速ドライブ（対称）

対称のS字加減速ドライブを自動減速で行うには、WR3 レジスタの D2～0 ビットを次のように設定します。また、次のパラメータを設定する必要があります。

表 2.2-6 モード設定：S字加減速（対称）

モード設定ビット	記号	設定値	コメント
WR3/D0	MANLD	0	自動減速
WR3/D1	DSNDE	0	減速時に加速度設定値、加速度増加率設定値を使用する
WR3/D2	SACC	1	S字加減速

表 2.2-7 設定パラメータ：S字加減速（対称）

パラメータ名	記号	コメント
加速度増加率	JK	
加速度	AC	最大値：536,870,911（1FFF FFFFh）をセットする
初速度	SV	
ドライブ速度	DV	
移動パルス数／終点	TP	連続パルスドライブでは不要

■ S字加減速の三角波形防止

S字加減速ドライブにおいても、速度カーブの滑らかさを保つための三角波形防止機能を搭載しています。S字加減速ドライブの三角波形防止は、加速と減速が対称であるS字加減速の定量パルスドライブにおいて出力パルス数がドライブ速度までの加速で必要とするパルス数に満たない場合と、S字加速中に減速停止させた場合の両方で機能し、滑らかな速度カーブを失いません。

＜定量パルスドライブでの三角波形防止＞

加速と減速が対称であるS字加減速の定量パルスドライブにおいて、出力パルス数がドライブ速度までの加速に必要とするパルスに満たない場合、次のような方式をとっています。

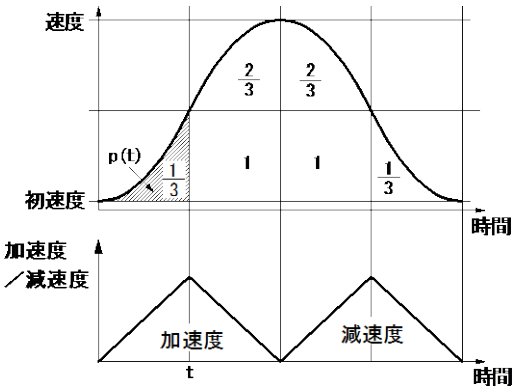


図 2.2-9　S 字加減速の 1/12 則

初速度を0としたとき、加速度をある加速度増加率で時間tまで増加させます。この加速度の増加区間で、時間tにおける速度v(t)は、次式で表せます。

$$v(t) = at^2$$　　a：速度に関する係数

よって、時間0からtの間に消費するパルス数 p(t)は、速度 v(t)を時間0からtまで積分した値であり、次式となります。

$$p(t) = \frac{1}{3} \times at^3$$

この値は、加速度増加率の値に関係なく、$at^2 \times t$（図中の一ますのパルス数）の 1/3 であることを表しています。

定量パルスドライブにおいて、時間0からtまで加速度をある加速度増加率で増加させ、時間tから同じ加速度増加率で加速度を減少させます。加速度が0になったら、減速時も同様に、同じ加速度増加率で減速度の増加／減少を行うと、定量パルスドライブ全体で消費されるパルス数は、図 2.2-9 に示すように、次式のパルス数になります。

$$\frac{1}{3} + \frac{2}{3} + 1 + 1 + \frac{2}{3} + \frac{1}{3} = 4\text{ます目分}$$

従って、ドライブ始めの加速度増加区間である時間0からtまでの消費パルス数（1/3 ます目）は、定量パルスドライブ全体で消費されるパルス数の 1/12 になります。

以上の理由により、本ICではS字加減速の定量パルスドライブにおいて、加速度増加時のパルスが総出力パルスの 1/12 より大きくなると、加速度減少に移行し、図 2.2-9 のような速度カーブを描くようにしています。[1/12 則]
この方式は、厳密には初速度 = 0のとき理想のカーブになります。初速度は、実際は0にはできませんので、図中の速度0から初速度までのパルス数が余ることになり、このパルス数はピーク速度時に出力されることになります。

＜減速停止での三角波形防止＞

直線加減速ドライブにおいて加速中に減速停止させたときは、速度カーブが三角波形となります。しかし、S字加減速ドライブでは速度カーブの滑らかさをあくまで重視するため、図 2.2-10 のように加速時に減速停止がかかった場合、すぐ減速に移行せず、加速度をいったん0まで減少させてから減速に移行します。

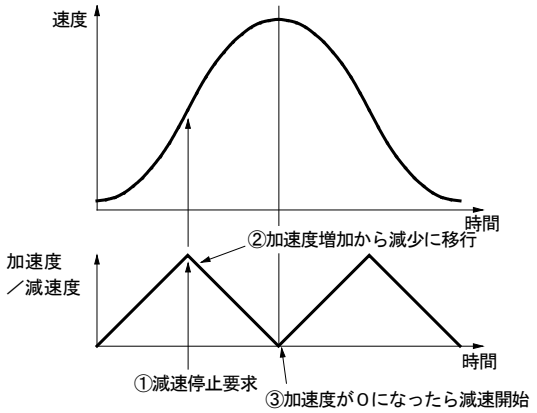


図 2.2-10 S字加減速の減速停止での三角波形防止

■ S字加減速ドライブの注意事項

a. S字加減速の定量パルスドライブにおいて、ドライブ速度をドライブ途中で変更することはできません。
b. S字加減速の定量パルスドライブにおいて、減速時に移動パルス数を変更すると正しいS字カーブを描くことができません。
c. S字加減速の定量パルスドライブでは、初速度を極端に低く設定すると、減速時に尻切れ（初速度まで落ちきる前に指定のドライブパルスを出し終えて終了する現象）や、引き摺り（初速度まで達してもまだ指定のドライブパルスを出し終えておらず、初速度で残りのドライブパルス出力する現象）が発生する場合があります。
d. S字加減速の連続パルスドライブにおいて、ドライブ途中の速度変更が可能です。
しかしながら、加減速途中においてはドライブ速度変更の命令は無効になり、速度変更はされません。
S字加減速の連続パルスドライブにおいて、速度変更をする場合には、必ず定速中（RR3 レジスタ ページ 1 CNST=1）に行ってください。
速度増加，減少命令（70h,71h）、および同期動作による速度変更も同様に無効になります。

■ パラメータ設定例（対称S字加減速）

初速度 100pps からドライブ速度 40kpps まで、0.4 秒でS字加速を行う例を示します。

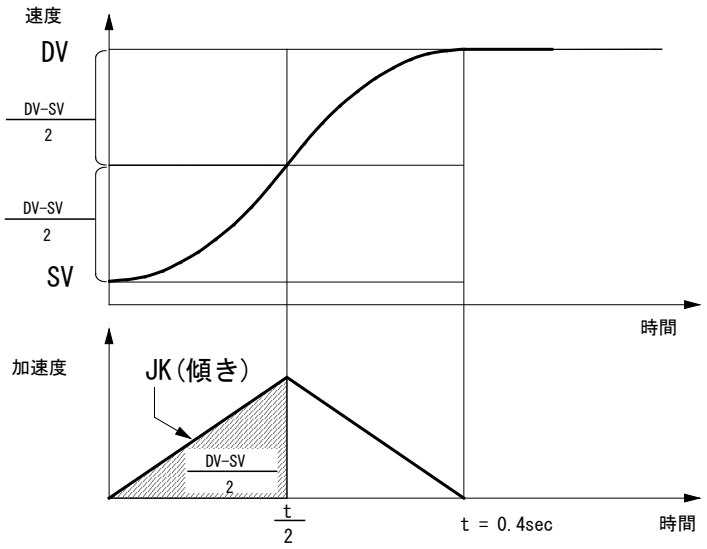


図 2.2-11　S字加減速ドライブ（対称）例

加速時には一定の加速度増加率(JK)に従って加速度を直線増加させていきます。この時、加速度直線を積分した値(斜線の面積)が初速度 SV からの速度の増加値になります。

加速時間全体(t = 0.4sec)の半分の時間(t/2)で、ちょうど速度が初速度(SV)からドライブ速度(DV)の半分の速度((DV-SV)/2)になるような加速度増加率(JK)を求めます。加速度増加率(JK)で増加する加速度を時間 0 から t/2 まで積分した値(斜線部の面積)は、時間 t/2 における速度の初速度からの上昇値に等しい事から、JK を求める式は次のようになります。

$$\frac{1}{2} \times JK \times \left(\frac{t}{2}\right)^2 = \frac{DV - SV}{2} \quad \text{より}$$

$$JK = \frac{4(DV - SV)}{t^2}$$

$$JK = \frac{4(40000 - 100)}{0.4^2} = 997,500 \ \text{pps/sec}^2$$

加速度増加率	JK [pps/sec^2]
ドライブ速度	DV [pps]
初速度	SV [pps]
加速時間	t [sec]

従って、図 2.2-11 に示した加速を持つ対称S字加減速ドライブのパラメータ設定は、以下のようになります。

モード設定	WR3 ← 0004h	WR3 レジスタのモード設定
加速度増加率	JK = 997500	
加速度	AC = 536870911	最大値（1FFF FFFFh）を設定
初速度	SV = 100	
ドライブ速度	DV = 40000	
移動パルス数	TP = 27500	定量パルスドライブのとき設定する

■ 部分S字加減速

加減速ドライブにおいて、直線の加速や減速区間をもち、加速や減速の開始と終了の部分だけS字の滑らかな速度カーブで加減速を行う事も可能です。速度パラメータの加速度、減速度の設定には最大値ではなく、直線加減速区間の加速度、減速度を指定します。図 2.2-12 の区間 b,f が直線加減速区間、区間 a,c,e,g がS字加減速区間になります。

区間 a において、加速度は 0 から指定の加速度設定値まで直線増加し、速度カーブは 2 次の放物線になります。指定の加速度に達すると加速度はその値を維持し、このとき速度カーブは区間 b の直線加速になります。指定のドライブ速度と現在速度の差が、加速度増加中に消費した速度分より少なくなると、加速度は指定の加速度増加率で減少し、区間 c の速度カーブは逆向きの放物線になります。減速時も同様に、部分S字減速カーブを生成します。

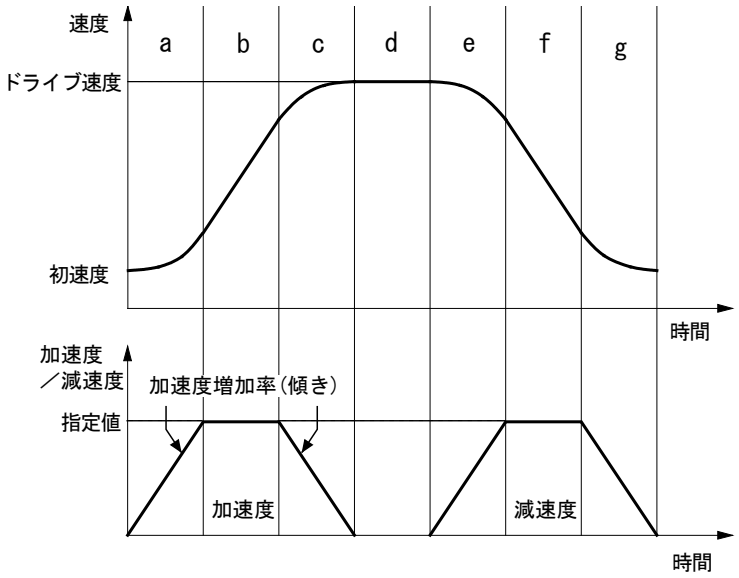


図 2.2-12 部分Ｓ字加減速ドライブ

■ パラメータ設定例（部分S字加減速）

下図のように、10kpps までを 0.2 秒で放物線加速し、10kpps から 30kpps までの 0.2 秒間を直線加速し、残りの 30kpps から 40kpps を 0.2 秒で放物線加速する部分S字加速の例を示します。

計算を単純にするために、初速度は 0 として無視します。
始めの 10kpps までの放物線加速において、加速度は 0.2 秒まで直線増加します。このときの加速度の積分値（斜線の面積）が、始めの放物線加速で立ち上げる速度 10kpps に相当します。
従って、0.2 秒時点の加速度は、10k×2／0.2＝100kpps/sec となり、加速度増加率は、100k／0.2＝500kpps/sec^2 となります。

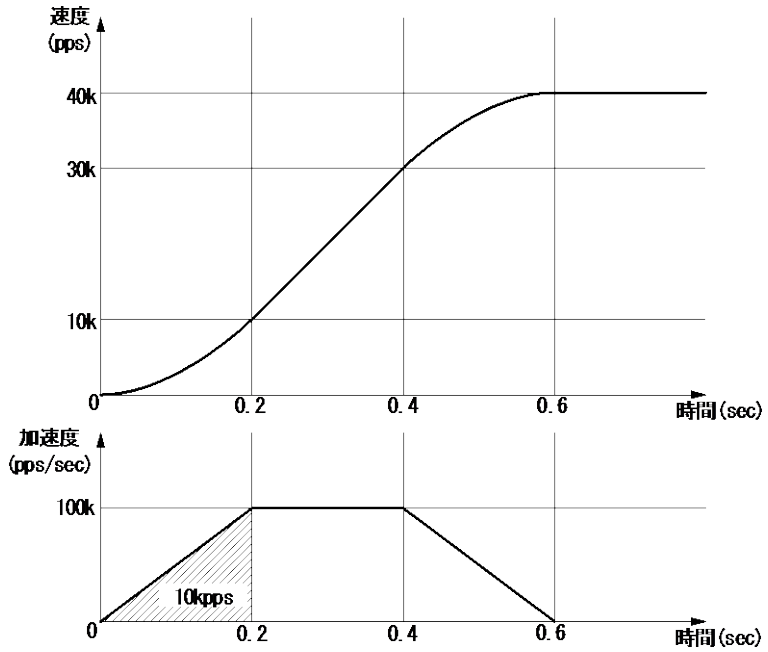


図 2.2-13 部分Ｓ字加減速ドライブ 例

実際には初速度 SV=0 には設定できないので、初速度 SV は 0 より大きい値を設定します。部分S字加減速の初速度 SV 設定値は、加速度 AC 設定値の平方根以上が目安です。
これにより、図 2.2-13 に示した加速を持つ対称部分S字加減速ドライブのパラメータ設定例は、以下のようになります。

モード設定	WR3 ← 0004h	WR3 レジスタのモード設定
加速度増加率	JK = 500000	放物線（S 字）加速区間の加速度増加率を設定
加速度	AC = 100000	直線加速区間の加速度を設定
初速度	SV = 400	
ドライブ速度	DV = 40000	
移動パルス数	TP = 40000	定量パルスドライブのとき設定する

2.2.5 非対称S字加減速

S字加減速ドライブにおいて、加速度増加率と減速度増加率を個別に設定することにより、非対称のS字カーブを作り出すことができます。ただし、非対称S字加減速の定量パルスドライブの場合は、対称S字加減速ドライブと異なり、自動減速できませんので、マニュアルで減速点を指定する必要があります。また、定量パルスドライブの三角波形防止機能(1/12 則)も働きませんので、加／減速度増加率、定量パルスドライブの出力パルス数に応じたドライブ速度を設定する必要があります。

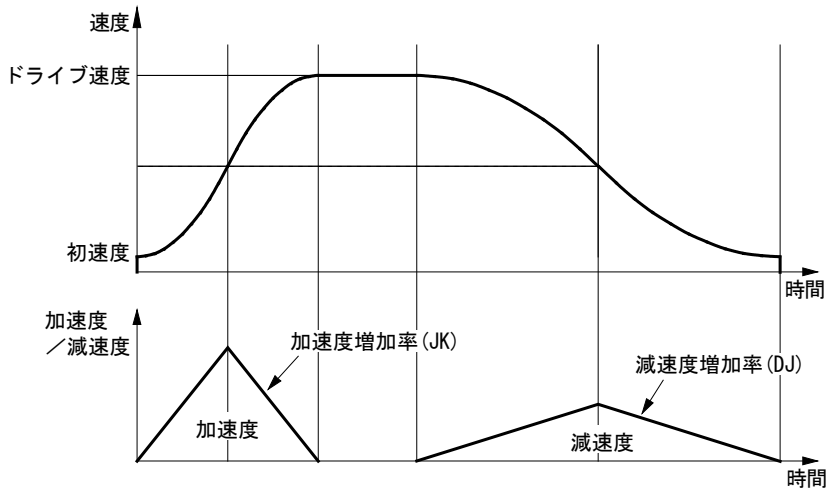


図 2.2-14 非対称Ｓ字加減速ドライブ

非対称S字加減速ドライブを行うには、WR3 レジスタの D2～0 ビットを次のように設定します。また、次のパラメータを設定する必要があります。

表 2.2-8 モード設定：非対称Ｓ字加減速

モード設定ビット	記号	設定値	コメント
WR3/D0	MANLD	1	マニュアル減速
WR3/D1	DSNDE	1	減速時に減速度設定値、減速度増加率設定値を使用する
WR3/D2	SACC	1	Ｓ字加減速

表 2.2-9 設定パラメータ：非対称Ｓ字加減速

パラメータ名	記号	コメント
加速度増加率	JK	
減速度増加率	DJ	
加速度	AC	最大値：536,870,911(1FFF FFFFh) をセットする
減速度	DC	最大値：536,870,911(1FFF FFFFh) をセットする
初速度	SV	
ドライブ速度	DV	
移動パルス数／終点	TP	連続パルスドライブでは不要
マニュアル減速点	DP	・定量ドライブの出力パルス数から 減速消費パルス数を引いた値を設定する ・連続パルスドライブでは不要

■ パラメータ設定例（非対称S字加減速）

加速時は初速度(SV)100pps からドライブ速度(DV)40kpps までを 0.2 秒で加速し、減速時はドライブ速度(DV)40kpps から初速度(SV)100pps までを 0.4 秒で減速する非対称S字加減速の例を示します。本ドライブは移動パルス数(TP)が 20,000 パルスの相対位置ドライブとします。

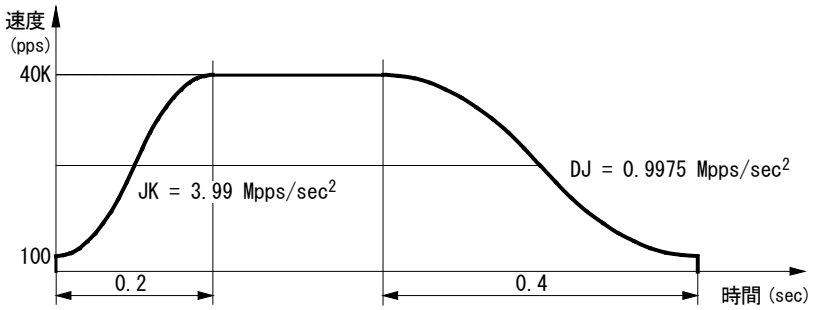


図 2.2-15 非対称Ｓ字加減速ドライブ 例

前述の対称S字加減速パラメータ設定例の式を使用して、加速度増加率、減速度増加率を求めます。

$$\text{加速度増加率} \quad JK = \frac{4(40000-100)}{0.2^2} = 3.99\,\text{Mpps/sec}^2$$

$$\text{減速度増加率} \quad DJ = \frac{4(40000-100)}{0.4^2} = 0.9975\,\text{Mpps/sec}^2$$

次に、非対称S字加減速では自動減速できませんので、マニュアルで減速点(DP)を設定します。マニュアル減速点は、定量ドライブにおいてドライブ開始から減速を開始する時点までの出力パルス数を設定します。相対位置ドライブでは移動パルス数(TP)から減速時消費パルス(Pd)を引いた値になりますので、まず減速時消費パルス(Pd)を求めます。

$$\text{減速消費パルス} \quad Pd = (DV+SV)\sqrt{\frac{DV-SV}{DJ}} = (40000+100)\sqrt{\frac{40000-100}{0.9975\times10^6}} = 8020$$

移動パルス数(TP)が 20,000 パルスの相対位置ドライブの減速時消費パルス(Pd)が 8,020 のとき、マニュアル減速点(DP)は次のようになります。

$$\text{マニュアル減速点} \quad DP = TP - Pd = 20000 - 8020 = 11980$$

従って、本 IC へのパラメータ設定は、以下のようになります。

モード設定	WR3←0007h	WR3 レジスタのモード設定
加速度増加率	JK = 3990000	
減速度増加率	DJ = 997500	
加速度	AC = 536870911	最大値 (1FFF FFFFh) を設定
減速度	DC = 536870911	最大値 (1FFF FFFFh) を設定
初速度	SV = 100	
ドライブ速度	DV = 40000	
移動パルス数	TP = 20000	
マニュアル減速点	DP = 11980	

【注意】上記の減速消費パルスを求める式は理想的な式となり、実際の IC ではパラメータの値によって、引き摺りや尻切れが発生します。

2.2.6 ドライブパルス幅と速度精度

■ ドライブパルスのパルス比率

＋方向／－方向のドライブパルスにおいて、ドライブ速度によって決まるパルス周期の時間は、演算上の誤差±1CLK (CLK=16MHz のとき±62.5nsec)はありますが、基本的には Hi レベルと Low レベルに 50%づつ振り分けられます。
例えば下図に示すように、DV = 1000pps に設定すると、ドライブパルスは、Hi レベル幅 = 500 μ sec、Low レベル幅 = 500 μ sec、周期 = 1.00 msec のパルスを出力します。

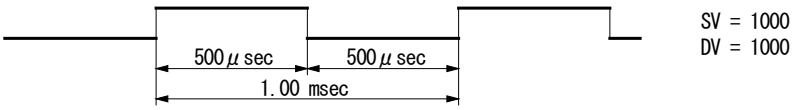


図 2.2-16 ドライブパルス出力の Hi/Low レベル幅 (1000pps)

加減速ドライブの加速時においては、1つのドライブパルスを出力している間にもドライブ速度は上昇していきますので、Low レベルのパルス幅が Hi レベルより短くなります。逆に、減速時においては、Low レベルのパルス幅が Hi レベルより長くなります。

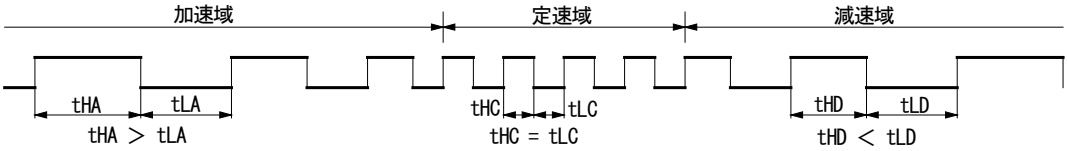


図 2.2-17 加減速ドライブ時のドライブパルス幅比較

■ ドライブ速度の精度

本ICでは、ドライブパルスを生成する回路は、すべて入力クロック信号(CLK)で動作しています。CLK 入力が、標準の 16MHz であれば、ある周波数のドライブパルスを生成しようとする場合、もし、ジッターのない均一な周波数のドライブパルスを作ろうとすると、CLK の周期の整数倍の周期を持った周波数しか作り出すことができません。例えば、2 倍:8.000 MHz、3 倍:5.333 MHz、4 倍:4.000 MHz、5 倍:3.200 MHz、6 倍:2.667 MHz、7 倍:2.286 MHz、8 倍:2.000 MHz、9 倍:1.778 MHz、10 倍:1.600 MHz、・・・・・・ の周波数しか出力することができず、これらの間の周波数を出力することができません。これでは任意のドライブ速度を設定することができなくなります。そこで、本ICでは、次の例に示すような方式により、任意のドライブ速度を出力するようにしています。

例えば、ドライブ速度 DV = 980kpps のドライブパルス出力について考えると、この周期は CLK の周期の整数倍ではないので、均一な周波数で 980kpps を出力することはできません。そこで、下図に示すように、CLK の周期に対して 16 整数倍の 1000kpps の周波数と 17 整数倍の 941kpps の周波数を合成して出力しています。980kpps の周期は、CLK(16MHz)の周期の 16.326 倍なので、CLK の 16 倍周期のパルスと 17 倍周期のパルスを 674:326 の比率で出力し、単位時間当たりの平均周期が CLK の 16.326 倍になるようにしています。

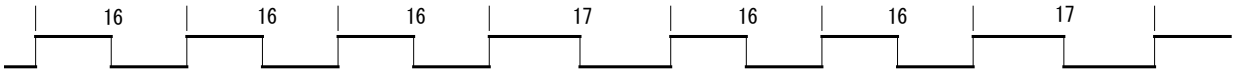


図 2.2-18 CLK 周期に対する 980kpps ドライブパルスの周期

この方式により、指定された速度のドライブパルスを精度良く出力することができます。指定した速度に対する実際に出力されるドライブパルスの速度精度は、±0.1%以下におさえています。

ドライブパルスをオシロスコープで観測すると、ドライブパルスの周期が CLK の周期の整数倍でないときには、上図のように、パルス周期に1CLK(62.5nsec)の時間差が生じますので、これがジッターのように見えますが、本ICはこの1CLK の時間差によって正しいドライブ速度を作り出しています。この1CLK の時間差は、モータを回す場合、負荷の慣性に吸収され、ほとんど問題になりません。

2.3 位置管理

本 IC は、現在位置を管理するための 32 ビットアップダウンカウンタを各軸2個(論理位置カウンタ、実位置カウンタ)持っています。多目的レジスタにあらかじめ値を設定し、現在位置との大小比較を行うことができます。また、論理位置カウンタ、実位置カウンタに対して、ソフトリミットや可変リング機能を設定することができます。

2.3.1 論理位置カウンタと実位置カウンタ

論理位置カウンタは、＋方向／－方向のドライブ出力パルスをIC内部でカウントします。＋方向1パルスで1カウントアップ、－方向1パルスで1カウントダウンします。
実位置カウンタはエンコーダなど外部からの入力パルスをカウントします。入力パルスを2相信号にするか、独立2パルス(カウントアップ／ダウン)信号にするかをモード選択することができます。2.12.3 項を参照してください。

両カウンタとも、ＣＰＵからのデータの書き込み／読み出しは常時可能です。カウント範囲は、-2,147,483,648～+2,147,483,647 です。負の値は2の補数で扱います。リセット時の論理位置カウンタ、実位置カウンタの値は不定です。

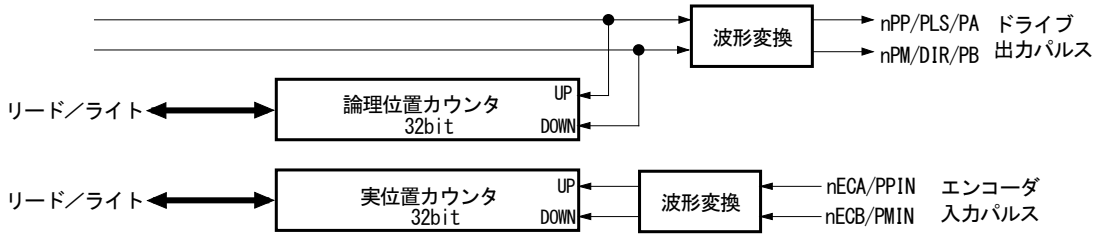


図 2.3-1 位置カウンタ部機能ブロック図

2.3.2 位置比較

本 IC は、各軸に論理位置カウンタ、実位置カウンタの現在位置と大小比較する多目的レジスタを 4 個持っています。ドライブ中においても多目的レジスタと論理／実位置カウンタとの大小関係をステータスで読み出すことができます。比較条件に合致したとき、信号を出力したり割り込み発生や同期動作起動を行うことができます。

多目的レジスタの比較機能については、2.4 節を参照してください。

2.3.3 ソフトリミット

各軸は論理位置カウンタ、または実位置カウンタに対して、ソフトリミットを設定することができます。ソフトリミットの設定対象はWR2 レジスタの D14 ビットで設定します。ソフトリミットを設定する2個の 32 ビットレジスタ(SLMT+, SLMT-)に対して、それぞれ＋方向／－方向のソフトリミット位置を設定します。
ソフトリミットの設定対象とした論理／実位置カウンタ値が SLMT+レジスタ値より大きくなると、ドライブは減速停止／即停止し、RR2 レジスタの D0 ビットに1が立ちます。このエラー状態は、－方向のドライブ命令を実行し、論理／実位置カウンタ値が SLMT+レジスタ値より小さくなることで解除されます。SLMT-レジスタの－方向についても同様です。
＋方向のソフトリミットは「位置カウンタ値≧SLMT+レジスタ値」のときソフトリミットエラーとなり、－方向のソフトリミットは「位置カウンタ値＜SLMT-レジスタ値」のときソフトリミットエラーになります。

図 2.3-2 は、ソフトリミット機能を有効にし、SLMT+レジスタ値=10000、SLMT-レジスタ値=-1000 を設定した例です。

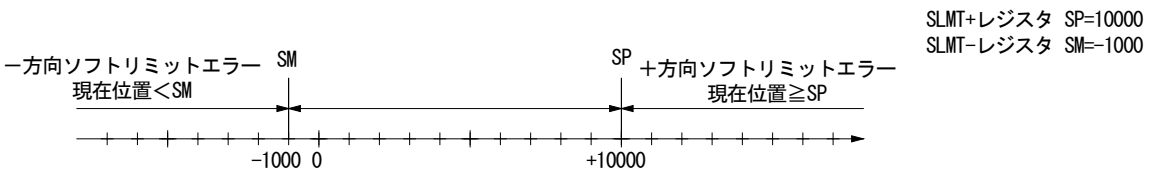


図 2.3-2 ソフトリミット値設定とソフトリミットエラー

ソフトリミット機能は有効／無効を設定できます。WR2 レジスタの D13 ビットで設定を行います。ソフトリミットの停止動作は減速停止／即停止から選択できます。WR2 レジスタの D15 ビットで設定を行います。SLMT+レジスタと SLMT-レジスタは常時書き込み可能です。
ソフトリミット機能は、リセット時は無効です。リセット時の SLMT+レジスタ、SLMT-レジスタの値は不定です。

2.3.4 位置カウンタの可変リング

論理位置カウンタおよび実位置カウンタは、32 ビット長のアップダウンリングカウンタです。従って通常は、32 ビット長の最大値である FFFF FFFFh から＋方向へカウントアップすると値が0に戻ります。また、0の値から－方向へカウントダウンすると FFFF FFFFh に戻ります。
可変リング機能はこのリングカウンタの輪の最大値を任意の値に設定する機能です。位置決め軸が直線運動ではなく、1回転すると元の位置に戻るような回転運動をする軸の位置管理をする場合に便利な機能です。

可変リングの大きさ、すなわち論理／実位置カウンタの最大値は、1～2,147,483,647（1～7FFF FFFFh）の範囲で任意の値に設定が可能です。可変リング機能は論理位置カウンタ最大値（LX）を論理位置カウンタ最大値書き込み命令（0Eh）で設定し、実位置カウンタ最大値（RX）を実位置カウンタ最大値書き込み命令（0Fh）で設定することで使用します。
リセット時の論理位置カウンタ最大値（LX）、実位置カウンタ最大値（RX）の値は FFFF FFFFh です。可変リング機能を使用しない場合は初期値のままにしておきます。

■ 可変リング設定例

10,000 パルスで1回転する回転軸の場合、次のように設定します。
① 論理位置カウンタ最大値（LX）に 9,999(270Fh)を設定。
② 実位置カウンタも使用する場合は、実位置カウンタ最大値（RX）に 9,999(270Fh)を設定。

このときのカウント動作
- ＋方向へカウントアップ時 ： …→9998→9999→0→1→…
- －方向へカウントダウン時 ： …→1→0→9999→9998→…

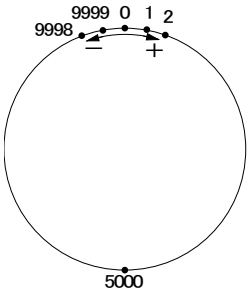


図 2.3-3 位置カウンタリング最大値 9999 の動作

【注意】
- 可変リング機能の位置カウンタの最大値として設定可能な値は、1～2,147,483,647（1～7FFF FFFFh）の範囲です。符号付き 32 ビットレジスタの負の値（8000 0000h～FFFF FFFEh）は設定できません。
- 論理位置カウンタ値（LP）、および実位置カウンタ値（RP）を設定するとき、それぞれ論理位置カウンタ最大値（LX），実位置カウンタ最大値（RX）の範囲外の値に設定することはできません。

2.4 多目的レジスタ

本 IC は、各軸に符号付き 32 ビットの多目的レジスタを 4 個(MR3～0)持っています。
多目的レジスタは、現在位置や速度などを比較対象とし、それらの値との比較に使用することができます。比較結果を信号出力したり、比較した大小関係の変化を同期動作の起動要因や、割り込み発生要因として使用することができます。また、同期動作の動作(Action)として、あらかじめ多目的レジスタに設定した値を新たな速度や移動パルス数として設定(ロード)したり、多目的レジスタに現在位置や現在速度を取得(セーブ)することができます。

多目的レジスタへのデータ書き込み／読み出しは常時可能で、4 個の多目的レジスタそれぞれの多目的レジスタ設定命令(10h～13h)、多目的レジスタ読み出し命令(34h～37h)によって行います。
リセット時の多目的レジスタの値は不定です。

2.4.1 比較対象と比較条件

多目的レジスタ MR3～0 の比較対象として、論理位置カウンタ値、実位置カウンタ値、現在速度値、現在タイマー値を設定できます。比較対象との比較条件式は ≧, >, =, < の 4 種類から選択できます。

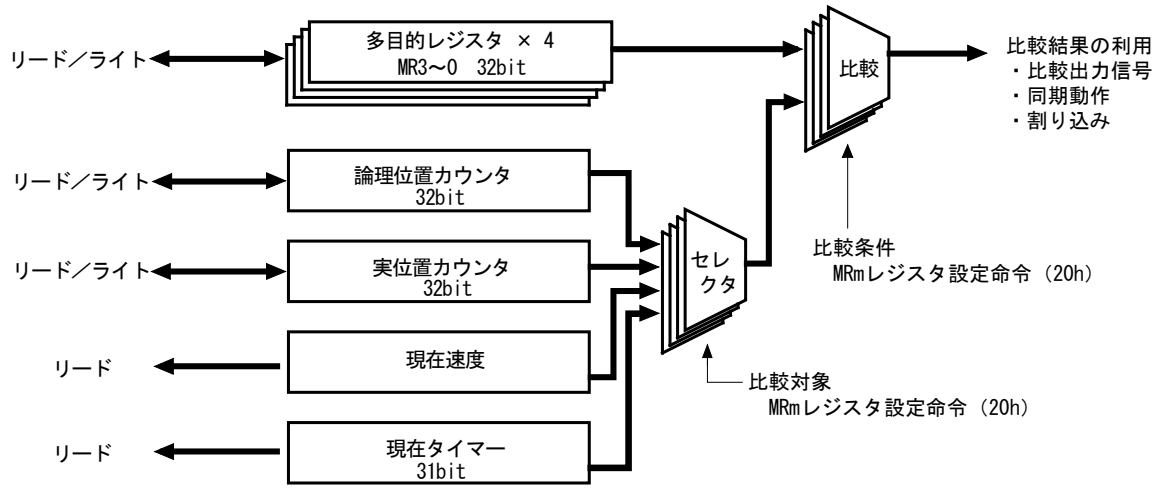


図 2.4-1 多目的レジスタと比較機能

4 個の多目的レジスタに対して、それぞれ個別に比較対象、および比較条件を選択し設定することができます。比較対象、比較条件の設定は多目的レジスタモード設定命令(20h)で行います。WR6 ライトデータレジスタの指定のビットを設定し、多目的レジスタモード設定命令コード(20h)を WR0 レジスタに書き込むと設定されます。
多目的レジスタモード設定状態は、多目的レジスタモード設定読み出し命令(40h)で読み出すことができます。

多目的レジスタモード設定命令 (20h)

	H								L							
WR6	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	M3C1	M3C0	M3T1	M3T0	M2C1	M2C0	M2T1	M2T0	M1C1	M1C0	M1T1	M1T0	M0C1	M0C0	M0T1	M0T0
	MR3 比較条件		MR3 比較対象		MR2 比較条件		MR2 比較対象		MR1 比較条件		MR1 比較対象		MR0 比較条件		MR0 比較対象	

表 2.4-1 比較対象の設定 (k:0～3)

MkT1 ビット	MkT0 ビット	MRm 比較対象
0	0	論理位置カウンタ (LP)
0	1	実位置カウンタ (RP)
1	0	現在速度値 (CV)
1	1	現在タイマー値 (CT)

表 2.4-2 比較条件の設定 (k:0～3)

MkC1 ビット	MkC0 ビット	MRm 比較条件
0	0	比較対象≧MRm
0	1	比較対象＞MRm
1	0	比較対象＝MRm
1	1	比較対象＜MRm

【注意】

比較対象を「現在速度値(CV)」、比較条件を「比較対象＝MRm」に設定しているとき、直線またはS字加減速ドライブで加速度, 減速度が 4,194,304(400000h)pps/sec を超える場合には、比較結果が真(アクティブ)にならないことがあります。
比較対象が「現在速度値(CV)」で加速度, 減速度がこの値以上になるときには、比較条件として「比較対象＝MRm」は使用せず、「比較対象≧MRm」など他の条件を使用してください。

■ 設定例: 論理位置カウンタとの比較

X 軸の論理位置カウンタ値が 500,000 以上のとき、比較結果が真となる比較設定は次のようになります。

```
WR6 ← A120h
WR7 ← 0007h      MR0 値 : 500,000 を設定          ⇦ 比較値を MR0 に設定
WR0 ← 0110h
WR6 ← 0000h      D3,D2 : 0,0  比較条件 : ≧
                 D1,D0 : 0,0  比較対象 : 論理位置カウンタ (LP)   ⇦ MR0 の比較対象、比較条件を設定
WR0 ← 0120h      多目的レジスタモード設定書き込み
```

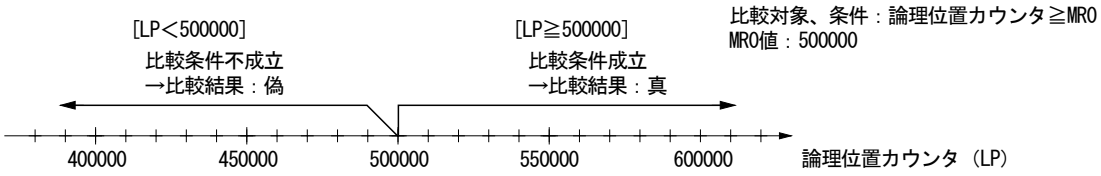


図 2.4-2 多目的レジスタと論理位置カウンタの比較 例

2.4.2 比較結果の用途

多目的レジスタと比較対象との比較結果は、比較出力信号、同期動作の起動要因、割り込み発生要因として利用できます。比較結果を利用する機能と、その動作について下表に示します。

表 2.4-3 比較結果の利用と動作

機能	対象	動作
比較出力信号	nPIO7～4 出力信号	比較結果が真のとき出力信号 Hi
同期動作の起動要因	同期動作 SYNC3～0	比較結果が真に変化したとき同期動作起動
割り込み発生要因	割り込み発生機能	比較結果が真に変化したとき割り込み発生

■ 比較出力信号

多目的レジスタの比較結果を、比較出力信号として出力することができます。多目的レジスタと比較対象の関係が、指定した比較条件を満たしているとき比較出力信号は Hi レベルを出力し、比較条件を満たしていないとき比較出力信号は Low レベルを出力します。

多目的レジスタ MR3～0 の比較出力は、それぞれ対応する比較出力信号 nPIO7～4 に出力されます。nPIO7～4 信号は、汎用入出力信号などほかの信号と兼用しています。これらを比較出力端子として使用するためには、あらかじめ PIO 信号設定1命令(21h)で、使用する nPIO7～4 信号の機能を比較出力信号に設定する必要があります。

表 2.4-4 多目的レジスタに対応する比較出力信号と設定ビット

多目的レジスタ	比較出力信号	PIO 信号設定 1 命令 (21h) WR6 レジスタの設定ビット
MR0	nPIO4	WR6/D9 ,8 : 1,1
MR1	nPIO5	WR6/D11,10 : 1,1
MR2	nPIO6	WR6/D13,12 : 1,1
MR3	nPIO7	WR6/D15,14 : 1,1

汎用入出力 nPIOm 信号について詳細は、2.8 節を参照してください。

■ 設定例 比較出力信号

X軸のドライブ中、現在速度が 5,000pps を超えているとき XPIO5 出力信号に Hi を出力し、5,000pps 以下のとき XPIO5 出力信号に Low を出力する設定を行います。

```
WR6 ← 1388h
WR7 ← 0000h     MR1 値 : 5,000 を設定                    ⇦ 比較値を MR1 へ設定
WR0 ← 0111h
WR6 ← 0060h     D7,D6 : 0,1  比較条件 : >
                D5,D4 : 1,0  比較対象 : 現在速度 (CV)    ⇦ MR1 の比較対象、比較条件を設定
WR0 ← 0120h     多目的レジスタモード設定書き込み
WR6 ← 0C00h     D11,D10 : 1,1  XPIO5 機能 : MR1 比較出力
                                                         ⇦ XPIO5 信号の機能設定
WR0 ← 0121h     PIO 信号設定 1 書き込み
```

■ 同期動作起動

多目的レジスタの比較結果によって、同期動作を起動させることができます。多目的レジスタと比較対象の関係が、指定した比較条件を満たすように変化したとき同期動作が起動します。同期動作を有効にした時点ですでに比較条件を満たしている場合、その時点では同期動作は起動せず、いったん比較条件を満たさない状態になってから再び比較条件を満たすように変化したとき、同期動作が起動します。

多目的レジスタ MR3～0 比較による同期動作起動は、それぞれ対応する同期動作セット SYNC3～0 の起動要因として設定できます。多目的レジスタの比較結果を同期動作の起動要因として使用するためには、あらかじめ同期動作 SYNC0,1,2,3 設定命令(26h,27h,28h,29h)で、使用する同期動作セットの起動要因を「MRm 比較が真に変化」(起動要因コード:01h)に設定し、同期動作有効命令(81h～8Fh)で使用する同期動作セットを有効に設定する必要があります。

表 2.4-5 多目的レジスタに対応する同期動作セットと設定命令

多目的レジスタ	同期動作セット	起動要因を設定する 同期動作設定命令
MR0	SYNC0	同期動作 SYNC0 設定命令(26h)
MR1	SYNC1	同期動作 SYNC1 設定命令(27h)
MR2	SYNC2	同期動作 SYNC2 設定命令(28h)
MR3	SYNC3	同期動作 SYNC3 設定命令(29h)

同期動作 SYNC0,1,2,3 設定命令では、起動要因以外にも動作や繰り返しの設定などの同期動作に関する設定を行います。同期動作の機能や設定について詳細は、2.6 節を参照してください。

■ 設定例 同期動作起動

X 軸に対し、10 秒タイマーの途中、タイマー開始 5 秒後に同期動作 SYNC2 で相対位置ドライブを起動する設定を行います。
この設定によってタイマーは、タイマー開始 5 秒後に同期動作を起動し、10 秒後にタイムアップするタイマーになります。

```
WR6 ← 4B40h
WR7 ← 004Ch     MR2 値 : 5,000,000 を設定          ⇦ 比較値を MR2 へ設定
WR0 ← 0112h      (5 秒＝5,000,000μsec)
WR6 ← 9680h
WR7 ← 0098h     タイマー値 : 10,000,000 を設定      ⇦ タイマー値を 10 秒に設定
WR0 ← 0116h      (10 秒＝10,000,000)
WR6 ← 0300h     D11,D10 : 0,0  比較条件 : ≧
                D9,D8 : 1,1  比較対象 : 現在タイマー値 (CT)   ⇦ MR2 の比較対象、比較条件を設定
WR0 ← 0120h     多目的レジスタモード設定書き込み
WR6 ← 00A1h     起動要因コード 01h : MR2 比較が真に変化
                動作コード 0Ah : 相対位置ドライブ起動        ⇦ 同期動作 SYNC2 の機能設定
WR0 ← 0128h     同期動作 SYNC2 設定書き込み
WR0 ← 0184h     同期動作 SYNC2 有効設定命令          ⇦ 同期動作 SYNC2 を有効に設定
```

※ 相対位置ドライブのためのパラメータはあらかじめ設定しておく必要があります。
相対位置ドライブについては 2.1.1 項を参照してください。

■ 割り込み発生

多目的レジスタの比較結果によって、割り込みを発生させることができます。多目的レジスタと比較対象の関係が、指定した比較条件を満たすように変化したとき割り込みが発生します。割り込みを許可した時点で、すでに比較条件を満たしている場合、その時点では割り込みは発生せず、いったん比較条件を満たさない状態になってから再び比較条件を満たすように変化したとき、割り込みが発生します。

多目的レジスタ MR3～0 比較により割り込みを発生させるためには、あらかじめ WR1 モードレジスタ 1 の各割り込み要因の許可／禁止ビットに対して、使用する多目的レジスタ比較の割り込みを許可に設定する必要があります。割り込みが発生した際の割り込み発生要因の確認は RR1 ステータスレジスタ 1 の各割り込み発生要因ビットで確認します。

表 2.4-6 多目的レジスタに対応する比較割り込み許可、確認ビット

多目的レジスタ	割り込み許可設定ビット	割り込み発生要因確認ビット
MR0	WR1／D0 : 1	RR1／D0 : 1
MR1	WR1／D1 : 1	RR1／D1 : 1
MR2	WR1／D2 : 1	RR1／D2 : 1
MR3	WR1／D3 : 1	RR1／D3 : 1

割り込みについて詳細は、2.10 節を参照してください。

■ 設定例 割り込み発生

X 軸に対し、実位置カウンタ値が 30,000 を通過すると割り込みが発生する設定を行います。

```
WR6 ← 7530h
WR7 ← 0000h     MR3 値 : 30,000 を設定              ⇦ 比較値を MR3 へ設定
WR0 ← 0112h
WR6 ← 0060h     D15,D14 : 1,0  比較条件 : ＝
                D13,D12 : 0,1  比較対象 : 実位置カウンタ (RP)  ⇦ MR3 の比較対象、比較条件を設定
WR0 ← 0120h     多目的レジスタモード設定書き込み
WR0 ← 011Fh     Ｘ軸指定(ＮＯＰ命令)                ⇦ 割り込み発生要因の設定
WR1 ← 0008h     割り込み発生要因 : MR3 比較が真に変化を許可
```

2.4.3 同期動作によるパラメータ値のロード／セーブ

同期動作を使用することによって、あらかじめ多目的レジスタに設定した値を新たな速度や移動パルス数として設定したり(ロード)、多目的レジスタに現在位置や速度を取得(セーブ)したりすることができます。

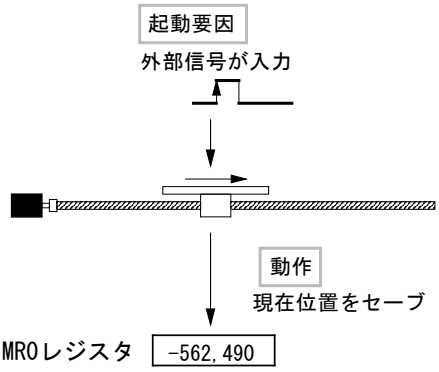


図 2.4-3 パラメータ値のセーブ 使用例

同期動作によって多目的レジスタからロード可能なパラメータ値は 7 種類、多目的レジスタへセーブ可能なパラメータ値は 5 種類用意されています。パラメータ値のロード／セーブは、起動する同期動作SYNC3～0に対応した多目的レジスタMR3～0に対して行われます。

同期動作によってパラメータ値のロード／セーブを行うためには、同期動作 SYNC0,1,2,3 設定命令(26h,27h,28h,29h)で、使用する同期動作セットの動作に各パラメータ値をロード／セーブするための動作コードを設定する必要があります。また、同期動作有効命令(81h～8Fh)で使用する同期動作セットを有効に設定する必要があります。

表 2.4-7 同期動作によってロード／セーブ可能なパラメータ

動作コード (Hex)	パラメータ値の設定 (ロード)	
01	ドライブ速度 (DV)	
02	移動パルス数／終点 (TP)	
03	スプリットパルス設定１ (SP1)	
04	論理位置カウンタ (LP)	(SYNC0)
	実位置カウンタ(RP)	(SYNC1)
	初速度 (SV)	(SYNC2)
	加速度 (AC)	(SYNC3)
0F	移動パルス数 (TP) 設定、かつ相対位置ドライブ起動	
10	終点 (TP) 設定、かつ絶対位置ドライブ起動	

動作コード (Hex)	現在値の取得 (セーブ)	
05	論理位置カウンタ (LP)	
06	実位置カウンタ (RP)	
07	現在タイマー (CT)	
08	現在速度 (CV)	(SYNC0)
	現在加減速度 (CA)	(SYNC1)

動作コード(Hex) : 同期動作 SYNC0,1,2,3 設定命令のライトデータレジスタに設定するコード

同期動作を使用した多目的レジスタへのパラメータ値のロード／セーブについて詳細は、2.6 節を参照してください。

2.5 自動原点出し

本ICは、CPUの介在なしに、高速原点サーチ → 低速原点追込みサーチ → エンコーダZ相サーチ → オフセット移動などの一連の原点出しシーケンスを自動的に実行する機能を持っています。自動原点出しは、下表に示すステップ1からステップ4を順に実行します。各ステップについて実行／不実行の選択ができます。不実行が選択された場合にはそのステップは実行されず、次のステップに移ります。また、各ステップではサーチ方向、検出信号をモード設定します。ステップ1，4はドライブ速度に設定された高速速度でサーチ動作または移動が行われます。ステップ2，3は原点検出速度に設定された低速速度でサーチ動作が行われます。また、ステップ2，3では、信号検出時に nDCC(偏差カウンタクリア)出力や実位置・論理位置カウンタをクリアさせることが可能です。各ステップ終了時にステップ間タイマーを挿入することが可能です。

表 2.5-1 自動原点出しの内容

ステップ番号	動 作	サーチ速度	検出信号
ステップ１	高速原点サーチ	ドライブ速度（ＤＶ）	nSTOP0, nSTOP1, リミット のいずれかを選択
ステップ２	低速原点サーチ	原点検出速度（ＨＶ）	nSTOP1, リミット のいずれかを選択
ステップ３	低速Ｚ相サーチ	原点検出速度（ＨＶ）	nSTOP2
ステップ４	高速オフセット移動	ドライブ速度（ＤＶ）	なし

一般に、原点出しは使用する検出信号によって動作は様々です。次の例に示すように原点近傍信号と原点信号の 2 つのセンサを使用して行う場合や、原点信号単独、あるいは一方のオーバーランリミット信号のみを使用する方法があります。

(1) 原点近傍信号(nSTOP0)と原点信号(nSTOP1)を用いて行う場合の例
指定された方向に高速で原点近傍信号をサーチし、原点近傍信号を検出したら減速停止します。続いて低速で原点信号をサーチし、原点信号を検出したら即停止します。

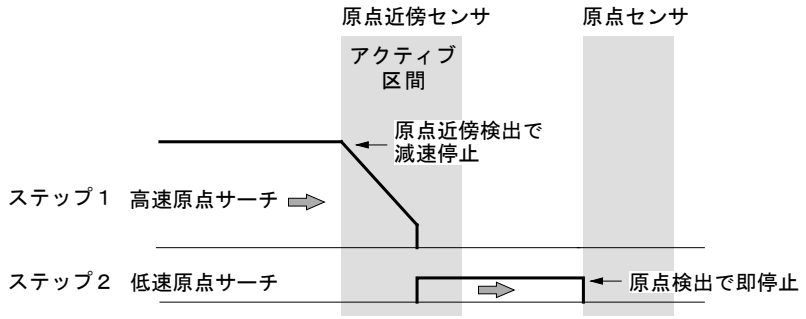
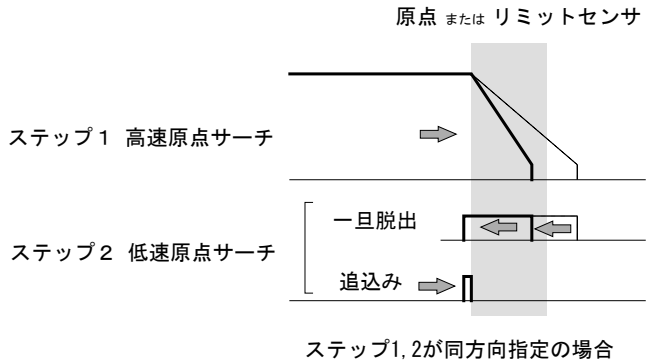


図 2.5-1 自動原点出し 例１

(2) 原点信号(nSTOP1)のみ、またはリミット信号(nLMTP/nLMTM)だけで行う場合の例
指定された方向に高速で原点信号またはリミット信号をサーチし、信号を検出したら減速停止します。続いて反対方向に信号のアクティブ区間から一旦脱出させます。その後、低速で原点信号をサーチし、原点信号を検出したら即停止します。検出信号にリミットを選択した場合は検出方向のリミット信号となります。



ステップ1,2が同方向指定の場合

図 2.5-2 自動原点出し 例２

本ICは、これらの様々な原点出しにも対応できるように、いくつかのモード設定が用意されています。

2.5.1 各ステップの動作

各ステップとも実行させるか否かを、また検出する＋／－方向を、また検出信号をモード設定で指定することができます。不実行に指定すると、そのステップは実行されないで次のステップに進みます。

■ ステップ1 高速原点サーチ

ドライブ速度（DV）に設定された速度で、指定の方向に、指定された検出信号がアクティブになるまでドライブパルスを出力します。検出信号はnSTOP0、nSTOP1、リミット信号から選択します。リミットを選択した場合は検出方向のリミット信号となります。高速サーチ動作を行わせるために、ドライブ速度（DV）を初速度（SV）より高い値に設定します。加減速ドライブが行われ、指定信号がアクティブになると減速停止します。

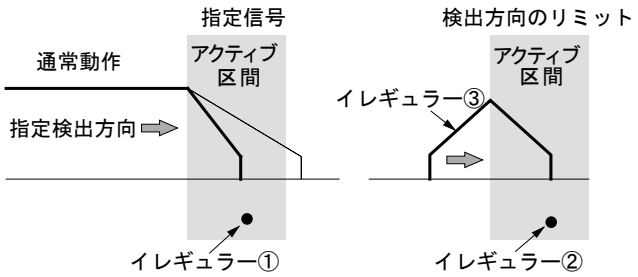


図 2.5-3 ステップ１の動作

イレギュラー動作

① ステップ1開始前にすでに指定の検出信号がアクティブになっている。 → ステップ2に進みます。

② 検出信号をnSTOP0、nSTOP1に指定している場合、ステップ1開始前に検出方向のリミット信号がアクティブになっている。 → ステップ2に進みます。

③ 検出信号をnSTOP0、nSTOP1に指定している場合、実行中に検出方向のリミット信号がアクティブになった。 → ドライブを停止してステップ2に進みます。

ステップ1のその他の動作

ステップ1終了時、ステップ間タイマーの起動をさせることができます。詳細は、2.5.3 項を参照してください。

【注意】ステップ1は高速サーチを行いますので、検出信号をリミット信号に指定する場合はリミット停止モードを減速停止(WR2／D12：1)に設定します。WR2 レジスタについては、6.6 節を参照してください。

■ ステップ2 低速原点サーチ

ステップ2の通常動作は、原点検出速度(HV)に設定された速度で、指定の方向に、指定された検出信号がアクティブになるまでドライブパルスを出力します。検出信号はnSTOP1、リミット信号から選択します。リミットを選択した場合は検出方向のリミット信号となります。低速サーチ動作を行わせるために、原点検出速度(HV)を初速度(SV)より低い値に設定します。定速ドライブが行われ、指定信号がアクティブになると即停止します。

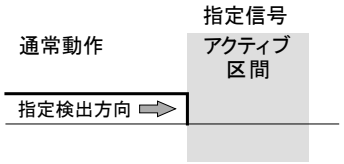


図 2.5-4 ステップ2の動作

イレギュラー動作

①ステップ2開始前にすでに指定信号がアクティブになっている。

[動作]指定信号が非アクティブになるまで、指定の検出方向と反対の方向へ原点検出速度(HV)で移動します。指定信号が非アクティブになったら、ステップ2の通常動作を始めから実行します。

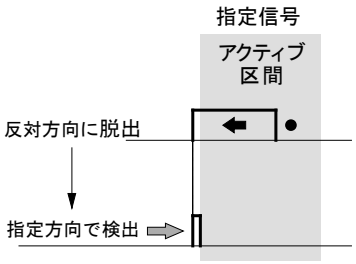


図 2.5-5 ステップ2 イレギュラー動作①

②検出信号がnSTOP1に指定されていて、ステップ2開始前に検出方向のリミット信号がアクティブになっている。

[動作]nSTOP1信号がアクティブになるまで指定の検出方向と反対の方向へドライブ速度(DV)で移動します。nSTOP1信号がアクティブになったら、さらにnSTOP1信号が非アクティブになるまで、指定の検出方向と反対の方向へ原点検出速度(HV)で移動します。nSTOP1信号が非アクティブになったら、ステップ2の通常動作を始めから実行します。

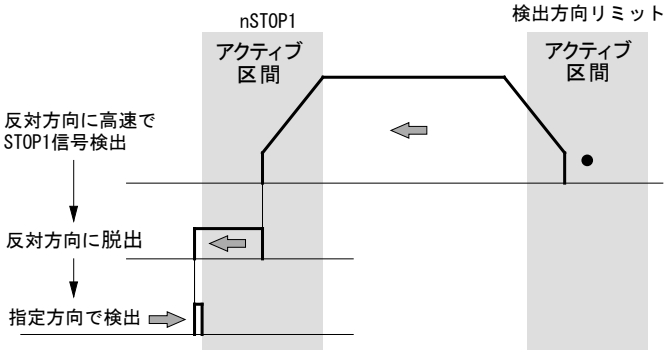


図 2.5-6 ステップ2 イレギュラー動作②

④ 検出信号がnSTOP1に指定されていて、実行中に検出方向のリミット信号がアクティブになった。

[動作]ドライブを停止して、イレギュラー②に示す動作を行います。

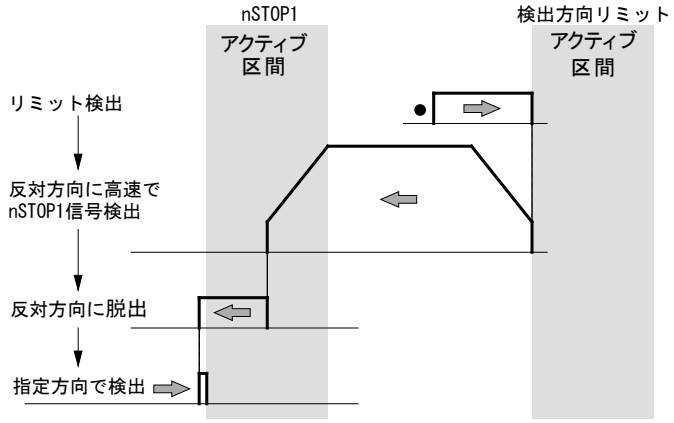


図 2.5-7 ステップ2 イレギュラー動作③

④ステップ1とステップ2の検出信号が同一で、かつステップ1とステップ2の検出方向が同一の場合、ステップ2開始前に指定信号が非アクティブになっている。
［動作］イレギュラー②に示す動作を行います。

この動作は回転軸の原点出しに適しています。

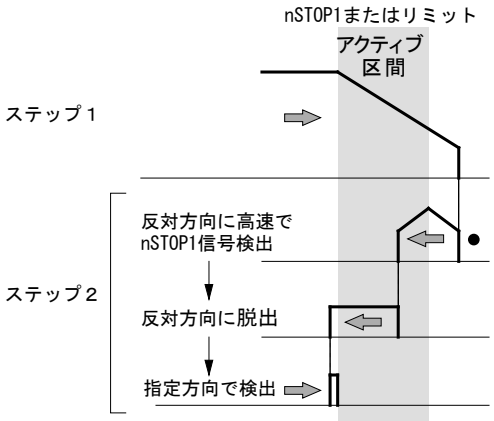


図 2.5-8 ステップ2 イレギュラー動作④

ステップ2のその他の動作
指定方向サーチ中に、ステップ2の検出信号が非アクティブからアクティブに変化したときに、偏差カウンタクリア(nDCC)信号を出力させる、また実位置カウンタ・論理位置カウンタをクリアさせることができます。ただし、イレギュラー動作中、指定と逆方向に移動しているときに検出信号がアクティブに変化してもこれらは作動しません。偏差カウンタクリア(nDCC)出力に関しては、2.5.2項を参照してください。
また、イレギュラー動作①～④の反対方向に脱出後、およびステップ2終了時にステップ間タイマーの起動をさせることができます。

■ ステップ3 低速Z相サーチ
原点検出速度（HV）に設定された速度で、指定の方向に、エンコーダZ相信号（nSTOP2）がアクティブになるまでドライブパルスを出力します。低速サーチ動作を行わせるために、原点検出速度（HV）を初速度（SV）より低い値に設定します。定速ドライブが行われ、エンコーダZ相信号（nSTOP2）がアクティブになると即停止します。

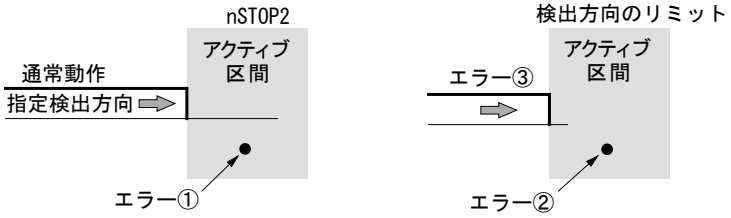


図 2.5-9 ステップ3の動作

検出条件として、エンコーダZ相信号（nSTOP2）と原点信号（nSTOP1）のAND条件で停止させることもできます。

ステップ3のその他の動作
エンコーダZ相信号（nSTOP2）がアクティブへ変化したときに、実位置カウンタ・論理位置カウンタをクリアさせることができます。実位置カウンタクリアは、nSTOP2 がアクティブになると CPU を介さないで実位置カウンタをクリアすることができます。これは、Z相をサーチするドライブ速度を低速にした場合にサーボ系あるいは機械系の遅れから発生するZ相検出の位置ずれが問題になるような場合に使用すると便利です。
同様に、エンコーダZ相信号（nSTOP2）がアクティブへ変化したときに、偏差カウンタクリア（nDCC)信号を出力させることができます。また、ステップ 3 終了時にステップ間タイマーの起動をさせることができます。

【注意】
① ステップ3開始時にすでにエンコーダZ相信号（nSTOP2）がアクティブになっているとエラーとなり、RR2 レジスタの D6 ビットに1が立ちます。自動原点出しは終了します。ステップ3は、必ずエンコーダZ相信号（nSTOP2）が安定した非アクティブ状態から開始するように、機械系を調整してください。
② ステップ3開始前に検出方向のリミット信号がアクティブになっているとエラーとなり、RR2 レジスタの検出方向のリミットエラービット（D2 または D3）に1が立ちます。自動原点出しは終了します。
③ 実行中に検出方向のリミット信号がアクティブになると検出動作は中断され、RR2 レジスタの検出方向のリミットエラービット（D2 または D3）に1が立ちます。自動原点出しは終了します。

■ ステップ4 高速オフセット移動

ドライブ速度(DV)に設定された速度で、移動パルス数(TP)に設定されているパルス数を相対位置ドライブでパルス出力します。このステップ4は、通常、機械的原点位置から作業原点に移動させたい場合に使用します。また、検出信号にリミット信号を選択した場合には、作業原点をリミットから少し離すために使用します。
ステップ4開始前、または実行中に移動方向のリミット信号がアクティブになるとエラー終了となり、RR2レジスタの検出方向のリミットエラービット(D2またはD3)に1が立ちます。自動原点出しは終了します。

2.5.2 偏差カウンタクリア出力

ステップ2またはステップ3動作時、指定された検出信号(ステップ3はnSTOP2に固定)がアクティブへ立ち上がるときに偏差カウンタクリア(nDCC)信号を出力させることができます。また、偏差カウンタクリア(nDCC)信号のパルス論理レベル、パルス幅を指定することができます。設定方法の詳細は、2.5.4項を参照してください。

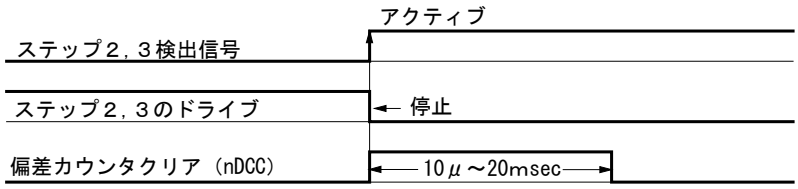


図 2.5-10 偏差カウンタクリア出力

偏差カウンタクリア出力は、ステップ2またはステップ3の検出動作終了と同時にアクティブになり、偏差カウンタクリア(nDCC)パルス出力の終了を待ってから次のステップが開始されます。

2.5.3 ステップ間タイマー

自動原点出しの各ステップは、モータ軸が逆転する設定があります。モータが急に逆転すると、機械系に大きな負荷が加わる場合があります。ステップ間タイマーは、この機械系に加わる負荷を軽減させるためのものです。

本ICでは、各ステップ終了時にステップ間タイマーを起動させることができます。ステップ2については、特定のイレギュラー動作後もステップ間タイマーを起動させることができます。
ステップ間タイマーの有無およびタイマー値を指定することができます。設定方法の詳細は、2.5.4項を参照してください。

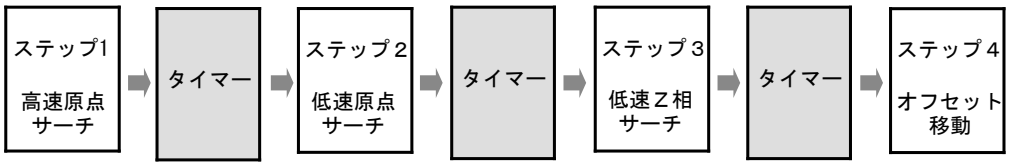


図 2.5-11 ステップ間タイマー

ステップ間タイマーを有効にした場合、各ステップ終了時にステップ間タイマーが起動し、タイマー動作後に次のステップが開始されます。ステップ2については、特定のイレギュラー動作が発生した場合は、そこでもステップ間タイマーが起動され、タイマー動作後にステップ2の通常動作が開始されます。ステップ2のイレギュラー動作に関しては、2.5.1項を参照してください。

【注意】ステップ間タイマーは各ステップごとに設定することはできません。有効にした場合、各ステップ間およびステップ2の特定のイレギュラー動作後の全てのステップ間タイマーが有効になり、指定したタイマー値のステップ間タイマーが起動されます。無効に設定した場合は、全てのステップ間タイマーは無効になります。

2.5.4 サーチ速度とモードの設定

自動原点出しを行わせるためには、次に記述する速度パラメータとモード設定が必要です。

■ 速度パラメータの設定

表 2.5-2 速度パラメータの設定

速度パラメータ	命令コード(hex)	説　　明
ドライブ速度（ＤＶ）	０５	ステップ１，４の高速でサーチ、移動を行う速度になります。 ただし、ステップ２のイレギュラー動作において、指定と逆方向で検出信号をサーチするときには、このドライブ速度になります。 加減速ドライブをさせるため、加速度(AC),初速度(SV)もあわせて適切な値に設定する必要があります。2.2.2項を参照してください。
原点検出速度（ＨＶ）	１４	ステップ２，３の低速でサーチを行う速度になります。 検出信号がアクティブになったとき即停止させるために、初速度（SV）より低い値に設定します。2.2.1項を参照してください。

■ 自動原点出しモード設定1

自動原点出しモード設定1は、下記のようにWR6レジスタの各ビットを設定し、自動原点出しモード設定1命令(23h)をWR0レジスタに書き込むとモード設定されます。各ステップの実行/不実行、検出信号の指定、検出方向および偏差カウンタクリア(nDCC)出力および論理位置カウンタ、実位置カウンタクリアの指定を行います。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6	S4EN	S3LC	S3RC	S3DC	S3DR	S3EN	S2LC	S2RC	S2DC	S2SG	S2DR	S2EN	S1G1	S1G0	S1DR	S1EN

① 各ステップの実行/不実行の設定

各ステップについて実行する場合は指定ビットに1を、不実行の場合は 0 を設定します。
各ステップの実行/不実行の指定ビットは下表のようになります。

表 2.5-3 各ステップの実行/不実行の指定ビット

	ステップ１	ステップ２	ステップ３	ステップ４	
実行/不実行	D0 ビット	D4 ビット	D10 ビット	D15 ビット	0:不実行
指定ビット	S1EN	S2EN	S3EN	S4EN	1:実行

② 各ステップの検出方向の設定

各ステップについて検出信号の検出方向が＋方向の場合は指定ビットに 0 を、－方向の場合は 1 を設定します。
各ステップの検出方向の指定ビットは下表のようになります。

表 2.5-4 各ステップの検出方向の指定ビット

	ステップ１	ステップ２	ステップ３	ステップ４	
検出方向	D1 ビット	D5 ビット	D11 ビット		0:＋方向
指定ビット	S1DR	S2DR	S3DR	―	1:－方向

③ 各ステップの検出信号の設定

ステップ1は nSTOP0、nSTOP1およびリミット信号から選択できます。ステップ2は nSTOP1 およびリミット信号から選択できます。
ステップ3は nSTOP2 信号固定となります。
ステップ1とステップ2は同じ信号を設定することが可能です。

ステップ1およびステップ2の検出信号の指定は下表のようになります。

表 2.5-5 ステップ１およびステップ２の検出信号の指定

ステップ１		
D3 ビット S1G1	D2 ビット S1G0	検出信号
0	0	nSTOP0
0	1	nSTOP1
1	0	リミット信号
1	1	（設定不可）

ステップ２	
D6 ビット S2SG	検出信号
0	nSTOP1
1	リミット信号

検出信号としてリミット信号を指定すると、ステップ1では D1 ビット(S1DR)で、ステップ2では D5 ビット(S2DR)で指定した検出方向側のリミット信号が選択されます。検出方向が＋方向の場合は nLMTP 信号、－方向の場合は nLMTM 信号となります。

検出する入力信号がHiアクティブにするかLowアクティブにするかの論理設定はWR2レジスタで行う必要があります。WR2レジスタについては、6.6節を参照してください。

④ 偏差カウンタクリア(nDCC)出力および実位置/論理位置カウンタクリアの設定

ステップ2と3では、指定された検出信号が非アクティブからアクティブへ立ち上がったときに偏差カウンタクリア(nDCC)信号を出力させるか否かを指定することができます。偏差カウンタクリア(nDCC)信号を出力させる場合は指定ビットに1を、出力させない場合は 0 を設定します。
また、ステップ2，3，4終了時、実位置カウンタ・論理位置カウンタをクリアさせることができます。
実位置カウンタ・論理位置カウンタをクリアさせる場合は指定ビットに1を、クリアさせない場合は 0 を設定します。

各ステップの偏差カウンタクリア(nDCC)出力、実位置カウンタ・論理位置カウンタのクリアの指定ビットは下表のようになります。

表 2.5-6 各ステップの nDCC 出力および実位置･論理置カウンタクリアの指定ビット

	ステップ１	ステップ２	ステップ３	ステップ４	
偏差カウンタクリア信号(nDCC)出力	—	D7 ビット S2DC	D12 ビット S3DC	—	0:出力させない 1:出力させる
実位置カウンタクリア	—	D8 ビット S2RC	D13 ビット S3RC	(※1)	0:クリアしない 1:クリアする
論理位置カウンタクリア	—	D9 ビット S2LC	D14 ビット S2LC	(※1)	

(※1)ステップ4終了時(ステップ4が実行設定時)の実位置カウンタ・論理位置カウンタクリアは、自動原点出しモード設定2命令(24h)の自動原点出し終了時のクリア設定で行うことになります。次の「■ 自動原点出しモード設定2」を参照してください。

■ 自動原点出しモード設定2

自動原点出しモード設定2は、下記のようにWR6レジスタの各ビットを設定し、自動原点出しモード設定2命令(24h)をWR0レジスタに書き込むとモード設定されます。偏差カウンタクリア(nDCC)出力のパルス論理およびパルス幅、ステップ間タイマーの有効/無効およびタイマー時間、自動原点出し終了時の実位置カウンタ・論理位置カウンタクリア、エンコーダZ相信号(nSTOP2)と原点信号(nSTOP1)のAND条件で停止の指定を行います。

	D15	D14	D13	D12 (H)	D11	D10	D9	D8	D7	D6	D5	D4 (L)	D3	D2	D1	D0
WR6						HTM2	HTM1	HTM0	HTME	DCP2	DCP1	DCP0	DCPL	LCLR	RCLR	SAND

① 偏差カウンタクリア(nDCC)出力のパルス論理とパルス幅の指定

各ステップで偏差カウンタクリア(nDCC)信号を出力させるとき、パルス論理とパルス幅を設定することができます。
パルス論理は D3 ビット(DCPL)で指定します。下図のように、0 を指定すると Hi パルス、1 を指定すると Low パルスとなります。

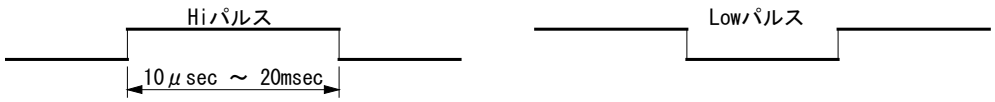


図 2.5-12 偏差カウンタクリア出力パルスの論理レベル

パルス幅は D6～4 ビット(DCP2～DCP0)の 3 ビットで指定します。
指定できるパルス幅は下表のようになります。

表 2.5-7 偏差カウンタクリア出力のパルス幅

WR6/D6 DCP2	WR6/D5 DCP1	WR6/D4 DCP0	パルス幅 (CLK=16MHz 時)
0	0	0	10 μsec
0	0	1	20 μsec
0	1	0	100 μsec
0	1	1	200 μsec
1	0	0	1 msec
1	0	1	2 msec
1	1	0	10 msec
1	1	1	20 msec

② ステップ間タイマーの指定

ステップ間タイマーの有効/無効およびタイマー時間を設定することができます。
有効/無効は D7 ビット(HTME)で指定します。1 を指定する有効、0 を指定すると無効となります。
タイマー時間は、D10～7 ビット(HTM2～HTM0)で指定します。
指定できるタイマー時間は下表のようになります。

表 2.5-8 ステップ間タイマーの時間幅の指定

WR6/D10 HTM2	WR6/D9 HTM1	WR6/D8 HTM0	タイマー時間 (CLK=16MHz 時)
0	0	0	1 msec
0	0	1	2 msec
0	1	0	10 msec
0	1	1	20 msec
1	0	0	100 msec
1	0	1	200 msec
1	1	0	500 msec
1	1	1	1000 msec

③ 自動原点出し終了時の実位置カウンタ・論理位置カウンタのクリア

自動原点出し終了時に、実位置カウンタ・論理位置カウンタのクリアを設定することができます。
実位置カウンタのクリアは D1 ビット(RCLR)で指定します。クリアさせる場合は 1 を、クリアさせない場合は 0 を設定します。
論理位置カウンタのクリアは D2ビット(LCLR)で指定します。クリアさせる場合は 1 を、クリアさせない場合は 0 を設定します。

④ エンコーダZ相信号(nSTOP2)と原点信号(nSTOP1)のAND条件で停止

ステップ3動作で、原点信号(nSTOP1)がアクティブ、かつ、エンコーダZ相信号(nSTOP2)がアクティブに変化したときにドライブを停止する機能です。D0 ビット(SAND)を 1 に設定すると、原点信号(nSTOP1)がアクティブ、かつ、エンコーダZ相信号(nSTOP2)がアクティブに変化したときに停止します。

【注意】この設定は、ステップ2の検出信号に nSTOP1 信号を設定したときのみ使用してください。ステップ2の検出信号にリミット信号を選択したときには、必ず 0 を指定してください。ステップ2の検出信号にリミット信号を選択していてこの設定を1にすると、正しい動作になりません。

2.5.5 自動原点出しの実行とステータス

■ 自動原点出しの実行

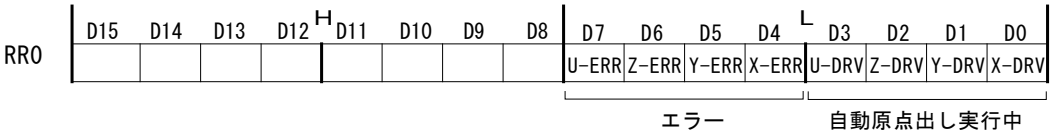
自動原点出しは、自動原点出し実行命令(5Ah)によって行います。自動原点出しモードと速度パラメータを正しく設定したのちに、WR0レジスタに命令コード5Ahを書き込むことにより開始されます。

■ 自動原点出しの中断

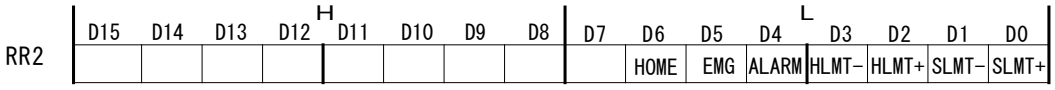
自動原点出しを途中で中断させたいときは、ドライブ減速停止命令(56h)、またはドライブ即停止命令(57h)を書き込みます。現在実行しているステップは中断されて、その後のステップは実行されず自動原点出しを終了します。
ステップ間タイマーが有効になっている時、ステップ間タイマー作動中に停止命令が発行された場合はステップ間タイマー動作も中断され、自動原点出しを終了します。

■ ステータスレジスタ

主ステータスレジスタRR0のD3～0ビット(n-DRV)はドライブ実行中を示すビットですが、自動原点出し実行時においても、自動原点出しを行っている軸のビットが実行中であることを示します。自動原点出しが開始されると、実行軸のビットが1になり、ステップ1動作開始からステップ4動作終了までの間、1を示しています。ステップ4を終了すると0に戻ります。



RR0レジスタのD7～4ビット(n-ERR)は自動原点出し中にエラーが発生すると実行軸のビットが1になります。エラー発生の要因は、次に示すRR2レジスタのD6～D0ビットに表示されます。



各々のエラー要因については、6.13節を参照してください。

RR3レジスタ ページ0のD14～9ビット(HSST5～0)は自動原点出しの実行ステートを番号で示します。自動原点出し実行中の現在実行している動作内容を知ることができます。

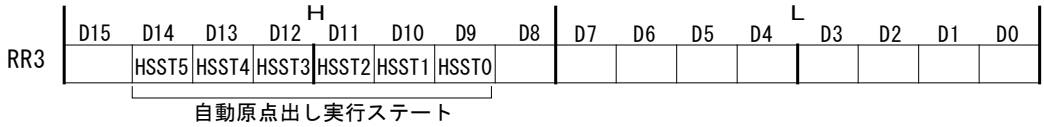


表 2.5-9 自動原点出しの実行ステータス

実行ステート	実行ステップ	動 作 内 容
0		自動原点出し実行命令待ち
3	ステップ１	指定検出方向で、検出信号のアクティブ待ち
6		ステップ１とステップ２間のタイマー動作中
１１	ステップ２	反指定検出方向で、検出信号のアクティブ待ち（イレギュラー動作）
１５		反指定検出方向で、検出信号の非アクティブ待ち （イレギュラー動作）
１８		イレギュラー動作後のタイマー動作中
２０		指定検出方向で、検出信号のアクティブ待ち
２３		ステップ２とステップ３間のタイマー動作中、または 偏差カウンタクリア出力中
２８	ステップ３	指定検出方向で、nSTOP2信号のアクティブ待ち
３２		ステップ３とステップ４間のタイマー動作中、または 偏差カウンタクリア出力中
３６	ステップ４	指定検出方向でオフセット移動中

2.5.6 自動原点出し時のエラー

自動原点出し実行中は、下表のようなエラーが発生する可能性があります。

表 2.5-10 自動原点出し時のエラー

エラー発生要因	エラー発生後のＩＣの動作	終了時の表示
ステップ１～４でnALARM信号がアクティブになった。	検出ドライブは即停止し、以降のステップは実行しないで終了する。	RR0/D7～4(実行軸)：１，RR2/D4：１
ステップ１～４でEMGN信号がアクティブになった。	検出ドライブは即停止し、以降のステップは実行しないで終了する。	RR0/ D7～4(実行軸)：１，RR2/D5：１
ステップ３で進行方向のリミット信号(nLMTP/M)がアクティブになった。（注）	検出ドライブは即停止/減速停止し、以降のステップは実行しないで終了する。	RR0/D7～4(実行軸)：１，RR2/D3またはD2：１
ステップ４で進行方向のリミット信号(nLMTP/M)がアクティブになった。（注）	オフセット移動は即停止/減速停止し、終了する。	RR0/D7～4(実行軸)：１，RR2/D3またはD2：１
ステップ３開始時にすでにnSTOP2信号がアクティブになっている。	以降のステップは実行しないで終了する。	RR0/D7～4(実行軸)：１，RR2/D6：１

自動原点出し終了後は、必ずRR0レジスタのD7～4ビット(n-ERR)を確認して下さい。自動原点出しを行った軸のエラービットに1が立っている場合は正しい自動原点出しが行われていません。

【注意】ステップ1，2では、進行方向の方向のリミットがアクティブになると減速停止もしくは即停止しますが、エラーにはなりません。

■ センサ故障時の症状

原点信号やリミット信号などのセンサ回路が定常的に故障した場合の症状について記述します。ただし、配線経路周囲のノイズや配線のゆるみ、素子の不安定動作などの原因による間欠的な故障については解析が難しく、ここでの記述内容には当たらない場合があります。また、これらの症状は、お客様のシステムの開発時において、信号レベルの論理設定を誤ったり、信号の配線を誤ったりしたときにも起きる場合があります。

表 2.5-11 センサ故障時の症状

故障要因		症状
リミットセンサの素子および配線経路の故障	常にＯＮのまま	その方向に動かず、終了時にリミットエラービット(RR2/D3またはD2)が１になっている。
	常にＯＦＦのまま	その方向の機械的終点にぶつかり、原点出し動作が終了しない。
ステップ１検出信号（nSTOP0,1）センサの素子および配線経路の故障	常にＯＮのまま	ステップ１を有効設定にし、信号がOFFの位置から自動原点出しを開始しているにもかかわらず、ステップ１（高速原点サーチ）を実行しないで、ステップ２に移ってしまう。
	常にＯＦＦのまま	ステップ１（高速原点サーチ）で、リミットをたたいて停止してからステップ２のイレギュラー動作に進む。 原点出しの結果は正しいが通常の動きではない。
ステップ２検出信号（nSTOP1を指定した場合）センサの素子および配線経路の故障	常にＯＮのまま	ステップ２（低速原点サーチ）で逆方向に動き出し、逆方向のリミットをたたいて停止する。終了時に逆方向リミットのエラービット(RR2/D3またはD2)が１になっている。
	常にＯＦＦのまま	ステップ２（低速原点サーチ）で指定方向のリミットをたたいてから、逆方向に移動をはじめて、逆方向のリミットをたたいて終了する。終了時に逆方向リミットのエラービット(RR2/D3またはD2)が１になっている。
Ｚ相（nSTOP2）センサの素子および配線経路の故障	常にＯＮのまま	ステップ３（低速Ｚ相サーチ）で、エラー終了する。RR2/D6が１になっている。
	常にＯＦＦのまま	ステップ３（低速Ｚ相サーチ）で、指定方向のリミットをたたいて停止する。終了時に指定方向リミットのエラービット(RR2/D3またはD2)が１になっている。

2.5.7 自動原点出しの注意点

■ サーチ速度

原点検出速度(HV)は、原点出し位置精度を上げるために、低速に設定する必要があります。入力信号がアクティブになったら即停止するように、初速度以下の値を設定します。
また、ステップ3のエンコーダZ相サーチを行う場合、Z相信号の遅延と原点検出速度(HV)の関係が重要です。例えば、Z相信号経路のフォトカプラの遅延時間とIC内蔵の積分フィルタ遅延時間の合計が、最大で500μsecかかるのであれば、エンコーダのZ相出力が1msec以上ONするように、原点検出速度を設定する必要があります。

■ ステップ3(Z相サーチ)開始位置

ステップ3のZ相サーチでは、Z相(nSTOP2)信号が、非アクティブ状態からアクティブに変化したときに検出ドライブを停止させます。従って、ステップ3の開始位置(すなわちステップ2の停止位置)が、安定してこの変化点から外れていなければなりません。通常は、ステップ3の開始位置がエンコーダのZ相位置の180°反対側になるように、機械的に調整します。

■ ソフトリミット

自動原点出し実行中は、ソフトリミットは無効にして下さい。ソフトリミットを有効にしておくと自動原点出しは正しく行われません。
自動原点出し正常完了後、論理位置カウンタ、実位置カウンタを正しく設定したのちに、ソフトリミットを設定して下さい。

■ 各入力信号の論理設定

自動原点出しで使用する入力信号(nSTOP0,1,2)のアクティブ論理設定は、WR2レジスタのビット(WR2/D0,D2,D4)で行います。
自動原点出しの時は、各信号を有効／無効にするビット(WR2/D1,D3,D5)の設定内容は無視されます。

2.5.8 自動原点出しの実例

■ 例１ 原点信号を用いた原点出し例

一つの原点信号で高速原点出しと低速での追い込みを行い、エンコーダのZ相検出は行わない例です。原点信号は必ずnSTOP1に入力します。

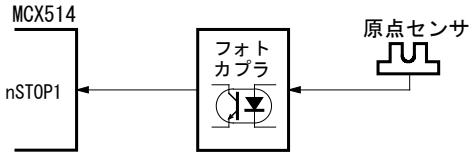


図 2.5-13 自動原点出し 例１の接続

自動原点出しの動作順序を下表に示します。

表 2.5-12 自動原点出し 例１の動作

ステップ	動作	実行/不実行	検出信号	信号レベル	検出方向	検出速度
1	高速サーチ	実行	nSTOP1	Low アクティブ	－方向	20,000pps
2	低速サーチ	実行			－方向	500pps
3	Ｚ相サーチ	不実行	－	－	－	－
4	オフセット移動	実行	－	－	＋方向	20,000pps

ステップ1では20,000ppsの高速速度で－方向にnSTOP1信号がLowを検出するまで移動し、Lowを検出したら減速停止します。ステップ2に移り、nSTOP1信号がLowレベル(アクティブ)であれば、イレギュラー動作①により500ppsの低速速度で指定方向と逆の方向(この場合は＋方向)に移動し、nSTOP1信号がHiレベルになる、すなわちnSTOP1アクティブ区間を脱出すると停止します。その後、500ppsの低速速度でステップ2の指定方向に移動し、再びnSTOP1信号がLowレベルになると停止します。

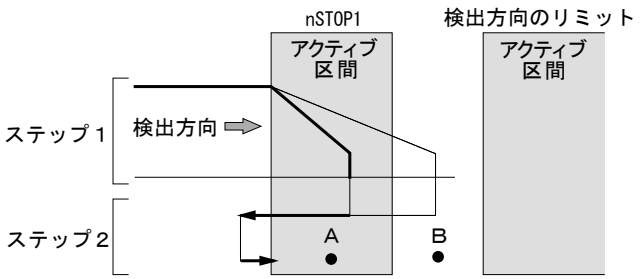


図 2.5-14 自動原点出し 例１の動作

ステップ1で、nSTOP1信号のアクティブ区間を通り過ぎて減速停止した場合でも、上図の破線に示すように一旦逆方向に戻り、nSTOP1アクティブ区間を脱出してから、ステップ2の指定方向で追い込みを行います。この動作は、ステップ1とステップ2の検出信号が同一で検出方向が同一の指定の場合に限ります。

自動原点出し開始位置が上図A点にある場合には、ステップ1は実行されず、ステップ2のイレギュラー動作①が行われます。また、図中B点にある場合には、ステップ1で検出方向のリミットをたたいてから、ステップ2のイレギュラー動作②が行われます。イレギュラー動作②については、2.5.1項を参照してください。

この例では、ステッピングモータ軸などエンコーダを使用しない場合の原点出しを想定して、ステップ3のZ相サーチは行わないものとします。ステップ4では、作業原点まで3500パルスだけ＋方向にオフセット移動させます。

【X軸のプログラム例】

```
// WR2レジスタ設定
WR0 ← 011Fh ﾗｲﾄ        // X軸選択
WR2 ← 0800h ﾗｲﾄ        // 原点信号論理設定： XSTOP1:Lowアクティブ
                       // ハードリミット有効

// 入力信号フィルタモード設定
WR6 ← 0A0Fh ﾗｲﾄ        // D11～D8  1010  フィルタ遅延:512μsec
                       // D2          1  XSTOP1信号：フィルタ有効
WR0 ← 0125h ﾗｲﾄ        // 命令書き込み

// 自動原点出しモード設定１
WR6 ← 8037h ﾗｲﾄ        // D15         1  ステップ４実行/不実行：    実行
                       // D14         0  ステップ３LPクリア         無効
                       // D13         0  ステップ３RPクリア         無効
                       // D12         0  ステップ３DCC出力：        無効
                       // D11         0  ステップ３検出方向：       －
                       // D10         0  ステップ３実行/不実行：    不実行
                       // D9          0  ステップ２LPクリア         無効
                       // D8          0  ステップ２RPクリア         無効
                       // D7          0  ステップ２DCC出力：        無効
                       // D6          0  ステップ２検出信号：       STOP1
                       // D5          1  ステップ２検出方向：       －方向
                       // D4          1  ステップ２実行/不実行：    実行
                       // D3,2      0,1  ステップ１検出信号：       STOP1
                       // D1          1  ステップ１検出方向：       －方向
                       // D0          1  ステップ１実行/不実行：    実行
WR0 ← 0123h ﾗｲﾄ        // 命令書き込み

// 自動原点出しモード設定２
WR6 ← 0000h ﾗｲﾄ        // D15         0
                       // D14         0
                       // D13         0
                       // D12         0
                       // D11         0
                       // D10～8      0  タイマー値
                       // D7          0  ステップ間タイマー         無効
                       // D6～4       0  DCCパルス幅
                       // D3          0  DCCパルス論理
                       // D2          0  原点出し終了時LPクリア     無効
                       // D1          0  原点出し終了時RPクリア     無効
                       // D0          0  ステップ２＆３             無効
WR0 ← 0124h ﾗｲﾄ        // 命令書き込み

// 高速原点サーチおよび低速原点サーチ速度の設定
WR6 ← 7318h ﾗｲﾄ        // 加減速度：95,000 PPS/SEC
WR7 ← 0001h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ        // 初速度：1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 4E20h ﾗｲﾄ        // ステップ１，４の速度：20000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 01F4h ﾗｲﾄ        // ステップ２の速度：500 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0114h ﾗｲﾄ

// オフセットパルスの設定
WR6 ← 0DACh ﾗｲﾄ        // オフセット移動パルス量 ：3500
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

// 自動原点出し実行開始
WR0 ← 015Ah ﾗｲﾄ
```

■ 例 2 リミット信号を用いた原点出し例

一方のリミット信号を原点信号として代用し、原点出しを行う方法です。ここでは、－方向リミット信号を原点信号として代用する例を示します。リミット信号を用いて原点出しを行う場合には、次の2項が条件となります。

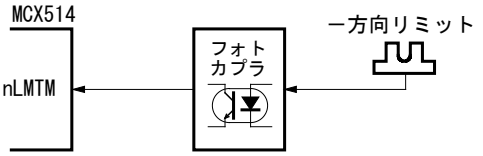


図 2.5-15 自動原点出し 例２の接続

a. ステップ1の高速検出動作を行う場合は、リミット信号がアクティブになる位置から機械的なリミットまでの距離内で、十分に減速停止できること。
b. 自動原点出しを開始する位置が、検出方向に向かって、リミット信号アクティブ区間を越えた先(図2.5-16のBの位置)にはないこと。

この例では自動原点出しの動作順序を下表のように行います。ステップ1,2のモード設定において、検出方向を－方向に指定し、検出信号をリミット信号に指定すると、－方向のリミット信号(nLMTM)が決まります。

表 2.5-13 自動原点出し 例２の動作

ステップ	動作	実行/不実行	検出信号	信号レベル	検出方向	検出速度
1	高速サーチ	実行	nLMTM	Low アクティブ	－方向	20,000pps
2	低速サーチ	実行			－方向	500pps
3	Z相サーチ	不実行	－	－	－	－
4	オフセット移動	実行	－	－	＋方向	20,000pps

ステップ1からステップ4までの動作は、前記の原点信号(nSTOP1)での場合と同様です。
自動原点出し開始位置が右図A点にある場合には、ステップ1は実行されず、ステップ2のイレギュラー動作①が行われ、一旦リミット信号アクティブ区間を逆方向に脱出してから指定方向の検出動作が行われます。

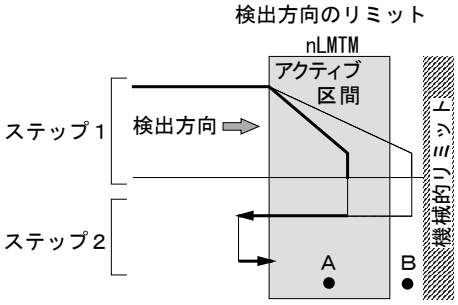


図 2.5-16 自動原点出し 例２の動作

【X軸のプログラム例】

```
// WR2レジスタ設定
WR0 ← 011Fh ﾗｲﾄ        // X軸選択
WR2 ← 1800h ﾗｲﾄ        // リミット信号論理設定： XLMTM:Lowアクティブ
                       // ハードリミット有効　減速停止          注1

// 入力信号フィルタモード設定
WR6 ← 0A0Fh ﾗｲﾄ        // D11～D8  1010  フィルタ遅延:512μsec
                       // D1          1  XLMTM信号：フィルタ有効
WR0 ← 0125h ﾗｲﾄ        // 命令書き込み

// 自動原点出しモード設定１
WR6 ← 807Bh ﾗｲﾄ        // D15         1  ステップ４実行/不実行：   実行
                       // D14         0  ステップ３LPクリア        無効
                       // D13         0  ステップ３RPクリア        無効
                       // D12         0  ステップ３DCC出力：       無効
                       // D11         0  ステップ３検出方向：      －
                       // D10         0  ステップ３実行/不実行：   不実行
                       // D9          0  ステップ２LPクリア        無効
                       // D8          0  ステップ２RPクリア        無効
                       // D7          0  ステップ２DCC出力：       無効
                       // D6          1  ステップ２検出信号：      LMTM
                       // D5          1  ステップ２検出方向：      －方向
                       // D4          1  ステップ２実行/不実行：   実行
                       // D3,2      1,0  ステップ１検出信号：      LMTM
                       // D1          1  ステップ１検出方向：      －方向
                       // D0          1  ステップ１実行/不実行：   実行
WR0 ← 0123h ﾗｲﾄ        // 命令書き込み
```

```
// 自動原点出しモード設定２
WR6 ← 0000h ﾗｲﾄ          // D15          0
                         // D14          0
                         // D13          0
                         // D12          0
                         // D11          0
                         // D10～8       0  タイマー値
                         // D7           0  ステップ間タイマー          無効
                         // D6～4        0  DCCパルス幅
                         // D3           0  DCCパルス論理
                         // D2           0  原点出し終了時LPクリア      無効
                         // D1           0  原点出し終了時RPクリア      無効
                         // D0           0  ステップ２＆３              無効
WR0 ← 0124h ﾗｲﾄ          // 命令書き込み

// 高速原点サーチおよび低速原点サーチ速度の設定
WR6 ← 7318h ﾗｲﾄ          // 加減速度：95,000 PPS/SEC
WR7 ← 0001h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ          // 初速度：1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 4E20h ﾗｲﾄ          // ステップ１，４の速度：20000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 01F4h ﾗｲﾄ          // ステップ２の速度：500 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0114h ﾗｲﾄ

// オフセットパルスの設定
WR6 ← 0DACh ﾗｲﾄ          // オフセット移動パルス量 ：3500
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

// 自動原点出し実行開始
WR0 ← 015Ah ﾗｲﾄ
```

注1：WR2レジスタのD10はリミット信号の論理、D11はリミット有効、D12はリミット動作を設定するビットですが、この例のように、リミット信号を検出信号に用いる場合、そのステップの動作では、D11の設定に関係なくリミット信号は有効になります(D11の設定情報は、リミット信号を検出信号に用いるステップでは動作に影響を与えません)。D12については必ず減速停止有効に設定して下さい。D10については使用状況に合わせて設定をして下さい。

【リミット信号使用時の注意】

ステップ1,2の検出方向は、必ず同じ方向とします。また、ステップ3(Z相サーチ)動作を行う場合はステップ1,2の方向とは逆の方向にします。ステップ4(オフセット移動)動作もステップ1,2の方向と逆の方向にし、必ずリミットアクティブ区間から脱出した所で自動原点出しを終了させるようにしてください。

■ 例 3 サーボモータ軸の原点出し例

パルス列入力のサーボモータの場合には、通常、ドライバ(サーボアンプ)からエンコーダのZ相信号が出力されています。高い位置精度の原点出しを行う場合には、このエンコーダZ相の出力タイミングに合わせてドライバ内部の偏差カウンタをクリアさせる必要があり、偏差カウンタクリア信号を入力しなければなりません。これらの信号を接続して原点出しを行う例を以下に示します。

下図のように、軸上に設置されている原点センサからインターフェイス回路を介して原点信号(nSTOP1)を入力します。サーボドライバにはエンコーダZ相入力(nSTOP2)、偏差カウンタクリア出力(nDCC)をインターフェイス回路を介して接続しています。

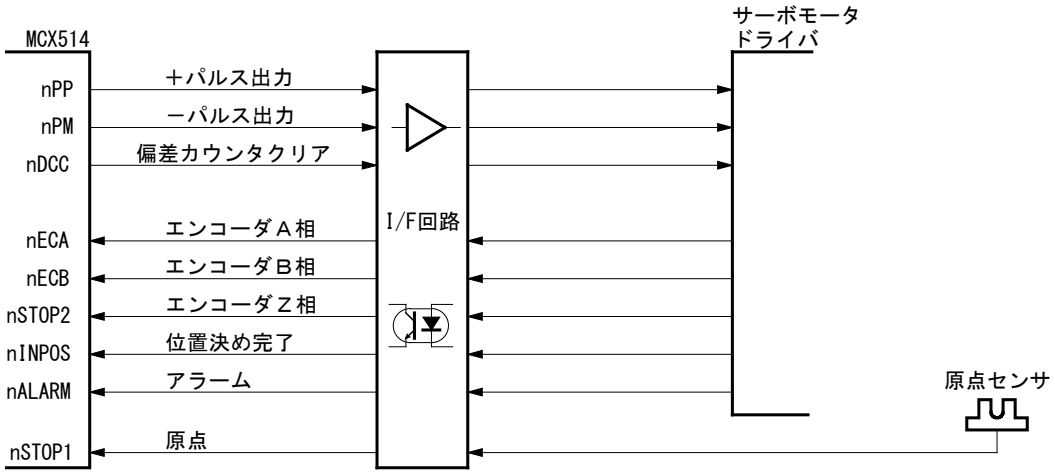


図 2.5-17 自動原点出し 例３の接続

【注意】エンコーダZ相入力は必ず本ICのSTOP2に接続します。速い応答性を必要としますので、インターフェイス回路には、ラインレシーバか高速フォトカプラが適しています。

表 2.5-14 自動原点出し 例３の動作

ステップ	動作	実行/不実行	検出信号	信号レベル	検出方向	検出速度
1	高速サーチ	実行	nSTOP1	Low アクティブ	－方向	20,000pps
2	低速サーチ	実行			－方向	500pps
3	Ｚ相サーチ	実行	nSTOP2	Low	－方向	500pps
4	オフセット移動	実行	－	－	＋方向	20,000pps

ステップ1からステップ2までの動作は、前記の原点信号(nSTOP1)の例の場合と同様です。ステップ2でnSTOP1入力がLowになるとステップ2が終了し、ステップ3に移行します。ステップ3では500ppsの速度で－方向にnSTOP2(Z相)信号がLowを検出するまで移動し、Lowを検出したら即停止します。nSTOP2入力信号の↓でnDCC(偏差カウンタクリア)を出力します。この例では、nDCC信号は100μsecの幅でHiパルスを出力するように設定しています。

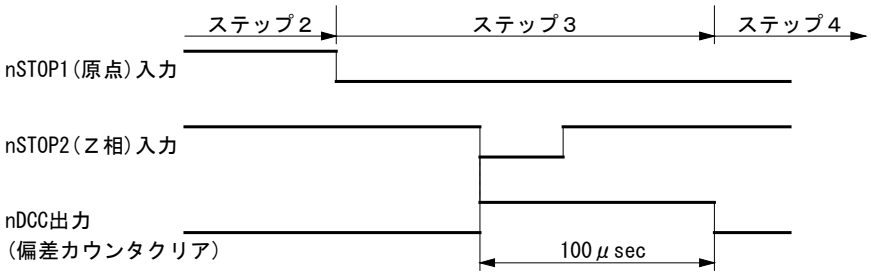


図 2.5-18 自動原点出し 例３の動作

また、ここでは、ステップ3においてnSTOP2(Z相)信号がLowアクティブになったときに実位置カウンタおよび論理位置カウンタをクリアさせるように設定します。

【X軸のプログラム例】

```
// WR2レジスタ設定
WR0 ← 011Fh ﾗｲﾄ          // X軸選択
WR2 ← 0800h ﾗｲﾄ          // 原点信号論理設定： XSTOP1,2:Lowアクティブ
                         // ハードリミット有効

// 入力信号フィルタモード設定
WR6 ← 0ACFh ﾗｲﾄ          // D15～D12 0000  フィルタFE6,7遅延:500nsec
                         // D11～D8  1010  フィルタFE0-5遅延:512μsec
                         // D6          1  XSTOP2信号：フィルタ有効
                         // D2          1  XSTOP1信号：フィルタ有効
WR0 ← 0125h ﾗｲﾄ          // 命令書き込み

// 自動原点出しモード設定１
WR6 ← FC37h ﾗｲﾄ          // D15         1  ステップ４実行/不実行：    実行
                         // D14         1  ステップ３LPクリア         有効
                         // D13         1  ステップ３RPクリア         有効
                         // D12         1  ステップ３DCC出力：        有効
                         // D11         1  ステップ３検出方向：       －方向
                         // D10         1  ステップ３実行/不実行：    実行
                         // D9          0  ステップ２LPクリア         無効
                         // D8          0  ステップ２RPクリア         無効
                         // D7          0  ステップ２DCC出力：        無効
                         // D6          0  ステップ２検出信号：       STOP1
                         // D5          1  ステップ２検出方向：       －方向
                         // D4          1  ステップ２実行/不実行：    実行
                         // D3,2      0,1  ステップ１検出信号：       STOP1
                         // D1          1  ステップ１検出方向：       －方向
                         // D0          1  ステップ１実行/不実行：    実行
WR0 ← 0123h ﾗｲﾄ          // 命令書き込み

// 自動原点出しモード設定２
WR6 ← 0020h ﾗｲﾄ          // D15         0
                         // D14         0
                         // D13         0
                         // D12         0
                         // D11         0
                         // D10～8      0  タイマー値
                         // D7          0  ステップ間タイマー         無効
                         // D6～4     010  DCCパルス幅                100μsec
                         // D3          0  DCCパルス論理              Hiパルス
                         // D2          0  原点出し終了時LPクリア     無効
                         // D1          0  原点出し終了時RPクリア     無効
                         // D0          0  ステップ２＆３             無効
WR0 ← 0124h ﾗｲﾄ          // 命令書き込み

// 高速原点サーチおよび低速原点サーチ速度の設定
WR6 ← 7318h ﾗｲﾄ          // 加減速度：95,000 PPS/SEC
WR7 ← 0001h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ          // 初速度：1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 4E20h ﾗｲﾄ          // ステップ１，４の速度：20000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 01F4h ﾗｲﾄ          // ステップ２，３の速度：500 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0114h ﾗｲﾄ

// オフセットパルスの設定
WR6 ← 0DACh ﾗｲﾄ          // オフセット移動パルス量 ：3500
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

// 自動原点出し実行開始
WR0 ← 015Ah ﾗｲﾄ
```

2.6 同期動作

本IC の同期動作は、各軸内、軸間、およびIC 外のデバイスとの間において、ドライブ途中に指定位置で外部信号を出力したり、反対に外部信号により現在通過位置を指定のレジスタにセーブするなどの様々な動作を連携して行う機能のことです。例えば次のような動作を行うことができます。

例1　　ドライブ中に指定位置を通過したら外部に信号を出力する。

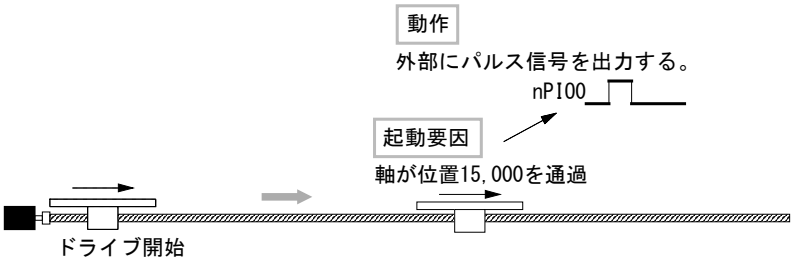


図 2.6-1　同期動作　例１

例2　　ドライブ中に外部から信号が入ったら、現在位置を所定のレジスタにセーブする。

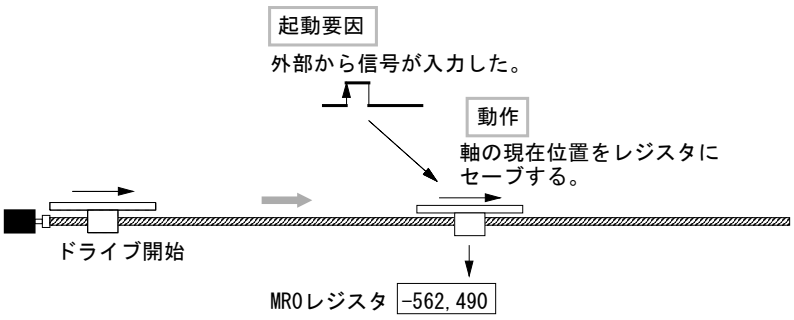


図 2.6-2　同期動作　例２

例3　　ドライブ中に指定位置から外部にスプリットパルスをN個、出力する。

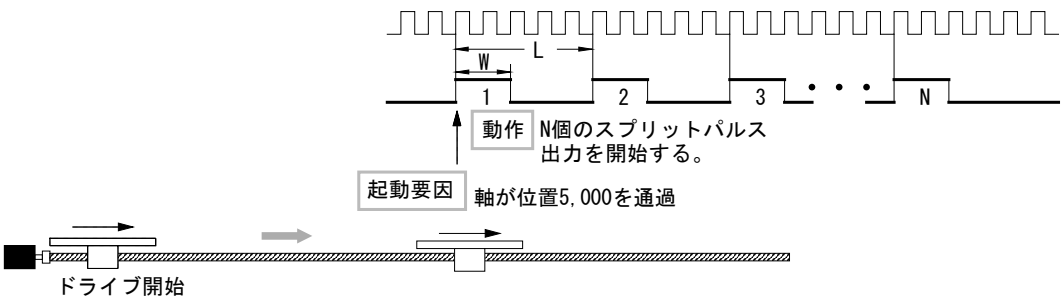


図 2.6-3　同期動作　例３

例4　　ドライブ中に、指定位置Aから指定位置Bまでを通過する時間を測定する。

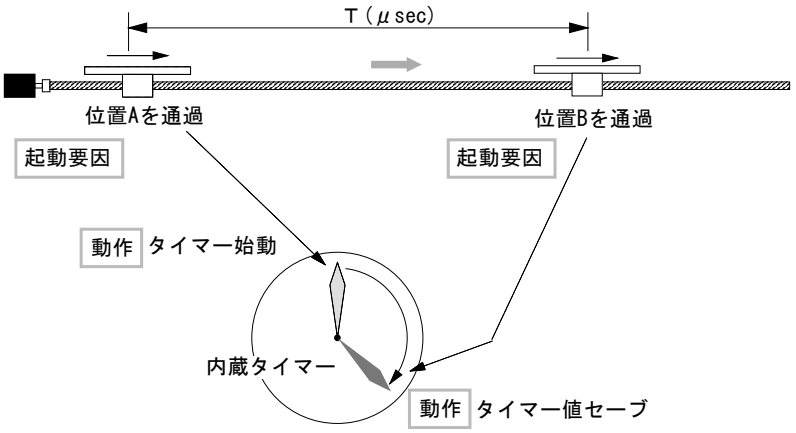


図 2.6-4　同期動作　例４

通常、このような同期動作は CPU 側でプログラムを組むことによっても行なうことはできます。しかし CPU の割り込み処理時間やプログラム実行時間による遅延が許されない様な場合に、本機能を使用すると便利です。本 IC の同期動作は、指定の起動要因が発生すると直ちに指定の動作を実行させる機能です。この連携動作は CPU の介在なしに行われ、精度の高い同期制御が可能となります。

指定の起動要因がアクティブになったら指定の動作を行わせることを1つの同期動作セットとすると、本ICは各軸に独立した4つの同期動作セットを持っています。また、各軸内のそれぞれ4つの同期動作セットを独立して動作させる以外に、4つの同期動作セットを連携して動作、軸間で連携して動作をさせることができます。

各軸の各々の同期動作セットSYNC0～3では、15種類の起動要因が用意されています。その中から1つを選択し、コード設定します。また、起動される動作(Action)は24種類用意されています。

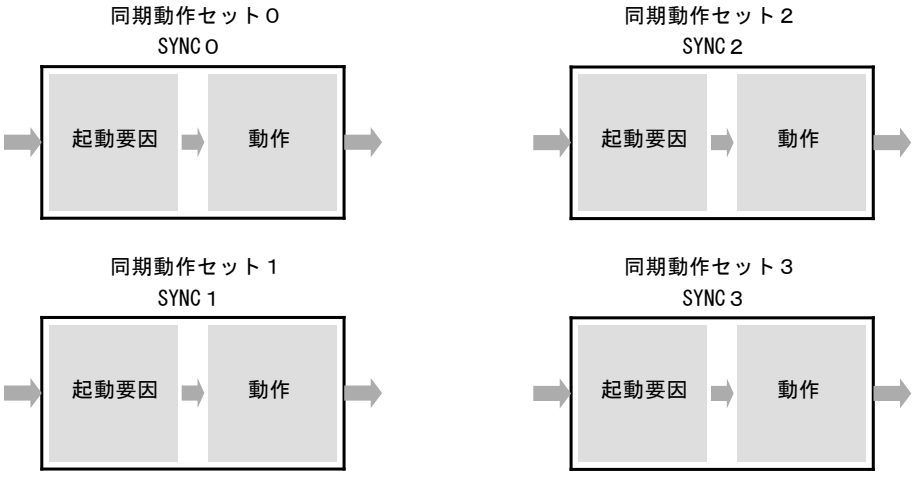


図 2.6-5　同期動作セット

2.6.1 起動要因

同期動作を起動する起動要因は、下表に示すように0hからFhのコードで指定する16個が用意されています。

表 2.6-1 起動要因一覧

設定コード (Hex)	同期動作セットO SYNC0	同期動作セット１ SYNC1	同期動作セット２ SYNC2	同期動作セット３ SYNC3	詳細
1	MR0対象が真に変化	MR1対象が真に変化	MR2対象が真に変化	MR3対象が真に変化	説明１
2	内蔵タイマーのタイムアップ				説明２
3	ドライブ開始				説明３
4	加減速ドライブの定速域が開始				説明３
5	加減速ドライブの定速域が終了				説明３
6	ドライブ終了				説明３
7	スプリットパルス開始				説明４
8	スプリットパルス終了				説明４
9	スプリットパルス出力				説明４
A	nPI00入力信号↑	nPI01入力信号↑	nPI02入力信号↑	nPI03入力信号↑	説明５
B	nPI00入力信号↓	nPI01入力信号↓	nPI02入力信号↓	nPI03入力信号↓	説明６
C	nPI04入力Low かつnPI00入力↑	nPI05入力Low かつnPI01入力↑	nPI06入力Low かつnPI02入力↑	nPI07入力Low かつnPI03入力↑	説明７
D	nPI04入力Hi かつnPI00入力↑	nPI05入力Hi かつnPI01入力↑	nPI06入力Hi かつnPI02入力↑	nPI07入力Hi かつnPI03入力↑	説明８
E	nPI04入力Low かつnPI00入力↓	nPI05入力Low かつnPI01入力↓	nPI06入力Low かつnPI02入力↓	nPI07入力Low かつnPI03入力↓	説明９
F	nPI04入力Hi かつnPI00入力↓	nPI05入力Hi かつnPI01入力↓	nPI06入力Hi かつnPI02入力↓	nPI07入力Hi かつnPI03入力↓	説明10
0	NOP				説明11

説明１： MRm対象が真に変化

多目的レジスタ(以下、MRmレジスタとします)の比較対象が比較条件を満たしたときに起動がかかります。表に示すように、4つの同期動作セットに対応するMRmレジスタは固定されています。比較対象と比較条件は多目的レジスタモード設定命令(20h)で設定します。例えば、MR0レジスタの比較対象には論理位置カウンタ(LP)が、比較条件には“比較対象≧MRm”が設定されている場合、論理位置カウンタの値がMR0の値と等しいか大きくなったときに起動がかかります。

同期動作を有効にしたときにすでに比較条件が真になっている場合には、一旦偽の状態になった後、ふたたび真になったときに同期動作が起動します。

説明２： 内蔵タイマーのタイムアップ

内蔵タイマーがタイムアップしたときに起動がかかります。タイマーの値はタイマー値設定命令(16h)で設定します。タイマーはタイマー始動命令(73h)の書き込み、または他の同期動作セットで開始させることができます。

説明３： ドライブ状態変化

下図に示すように、ドライブ中の速度状態の変化時に起動がかかります。

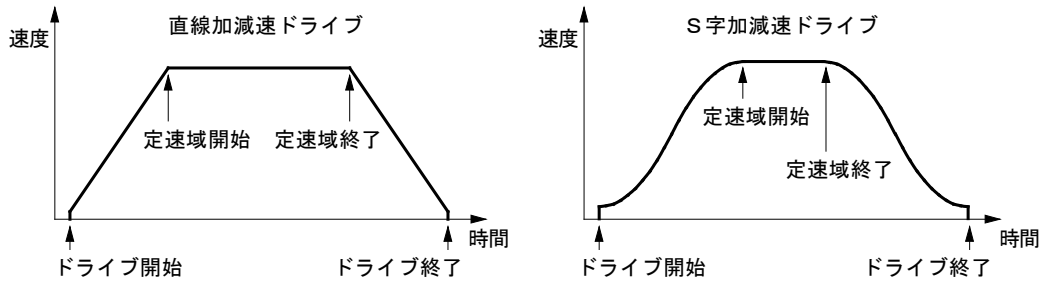


図 2.6-6 ドライブ状態に関する起動要因

【注意】加減速ドライブ中の定速域(一定速度でドライブが行われる区間)は、ドライブ終了時にも微小ながら発生する場合があります。

説明４： スプリットパルス

“スプリットパルス開始”は、スプリットパルス開始命令(75h)、あるいは他の同期動作セットでスプリットパルスが開始されたときに同期動作が起動します。
“スプリットパルス終了”は、最終のスプリットパルスを出力し終えたときに同期動作が起動します。
“スプリットパルス出力”は、スプリットパルスが出力される(有効レベルに立ち上がる、または立ち下がる)ときに同期動作が起動します。同期動作の設定を繰り返しにしておくと各々のスプリットパルス毎に同期動作が起動します。

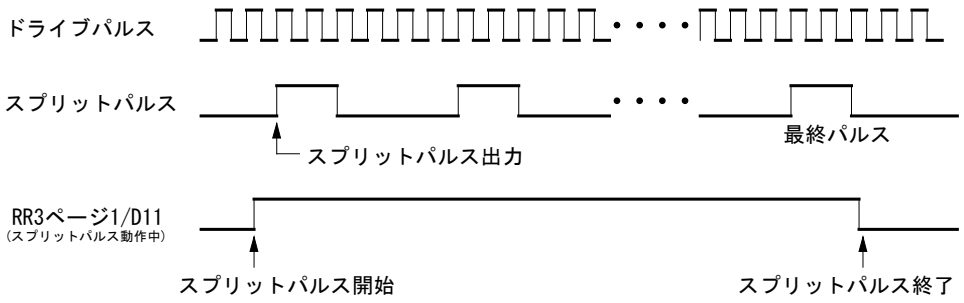


図 2.6-7 スプリットパルスの起動要因

説明５： 汎用入力信号の立ち上がり変化

“nPIOm入力信号↑”は、nPIOm (m=0～3)入力信号がLowレベルからHiレベルに立ち上がったときに起動がかかります。
表にあるように、4つの同期動作セットに対応するnPIOm信号は固定されています。
同期動作を有効にしたときにすでに入力信号がHiレベルになっている場合には、一旦Lowレベルに落ちた後、ふたたびHiレベルに立ち上がったときに同期動作が起動します。

説明６： 汎用入力信号の立ち下がり変化

“nPIOm入力信号↓”は、nPIOm (m=0～3)入力信号がHiレベルからLowレベルに立ち下がったときに起動がかかります。
表にあるように、4つの同期動作セットに対応するnPIOm信号は固定されています。
同期動作を有効にしたときにすでに入力信号がLowレベルになっている場合には、一旦Hiレベルに上がった後、ふたたびLowレベルに立ち下がったときに同期動作が起動します。

説明７： 汎用入力信号のLowレベルと立ち上がり変化

“nPIOm入力LowかつnPIOk入力↑“は、nPIOm (m=4～7)入力信号がLowレベルで、かつnPIOk (k=0～3)入力信号がLowレベルからHiレベルに立ち上がったときに起動がかかります。
表にあるように、4つの同期動作セットに対応するnPIOk、nPIOm信号は固定されています。
同期動作を有効にしたときにすでにnPIOm入力信号がLowでnPIOk入力信号がHiレベルになっている場合の動作は、説明5と同様です。

説明８： 汎用入力信号のHiレベルと立ち上がり変化

“nPIOm入力HiかつnPIOk入力↑”は、nPIOm (m=4～7)入力信号がHiレベルで、かつnPIOk (k=0～3)入力信号がLowレベルからHiレベルに立ち上がったときに起動がかかります。
表にあるように、4つの同期動作セットに対応するnPIOk、nPIOm信号は固定されています。
同期動作を有効にしたときにすでにnPIOm入力信号がHiでnPIOk入力信号がHiレベルになっている場合の動作は、説明5と同様です。

説明９： 汎用入力信号のLowレベルと立ち下がり変化

“nPIOm入力LowかつnPIOk入力↓”は、nPIOm (m=4～7)入力信号がLowレベルで、かつnPIOk (k=0～3)入力信号がHiレベルからLowレベルに立ち下がったときに起動がかかります。
表にあるように、4つの同期動作セットに対応するnPIOk、nPIOm信号は固定されています。
同期動作を有効にしたときにすでにnPIOm入力信号がLowでnPIOk入力信号がLowレベルになっている場合の動作は、説明6と同様です。

説明10： 汎用入力信号のHiレベルと立ち下がり変化

“nPIOm入力HiかつnPIOk入力↓”は、nPIOm (m=4～7)入力信号がHiレベルで、かつnPIOk (k=0～3)入力信号がHiレベルからLowレベルに立ち下がったときに起動がかかります。

表にあるように、4つの同期動作セットに対応するnPIOk、nPIOm信号は固定されています。

同期動作を有効にしたときにすでにnPIOm入力信号がHiでnPIOk入力信号がLowレベルになっている場合の動作は、説明6と同様です。

説明11： NOP

起動要因の条件を設定しない場合に用います。

例えば、モード設定で他SYNC起動を用いるとき、起動される同期動作セットの起動要因はNOPを設定します。

2.6.2 動作(Action)

下表に起動される動作(Action)を示します。コード01～09h,0Fh,10hは、同期動作セット0から同期動作セット4によって動作が異なります。

表 2.6-2 動作 (Action) 一覧

設定コード (Hex)	同期動作セット０ SYNC0	同期動作セット１ SYNC1	同期動作セット２ SYNC2	同期動作セット３ SYNC3	詳細
01	MR0 → DV	MR1 → DV	MR2 → DV	MR3 → DV	説明１
02	MR0 → TP	MR1 → TP	MR2 → TP	MR3 → TP	説明１
03	MR0 → SP1	MR1 → SP1	MR2 → SP1	MR3 → SP1	説明１
04	MR0 → LP	MR1 → RP	MR2 → SV	MR3 → AC	説明１
05	LP → MR0	LP → MR1	LP → MR2	LP → MR3	説明２
06	RP → MR0	RP → MR1	RP → MR2	RP → MR3	説明２
07	CT → MR0	CT → MR1	CT → MR2	CT → MR3	説明２
08	CV → MR0	CA → MR1	－	－	説明２
09	nPIO0信号パルス出力	nPIO1信号パルス出力	nPIO2信号パルス出力	nPIO3信号パルス出力	説明３
0A	相対位置ドライブ開始				
0B	反相対位置ドライブ開始				
0C	絶対位置ドライブ開始				
0D	＋方向連続パルスドライブ開始				
0E	－方向連続パルスドライブ開始				
0F	MR0値で相対位置ドライブ開始	MR1値で相対位置ドライブ開始	MR2値で相対位置ドライブ開始	MR3値で相対位置ドライブ開始	説明４
10	MR0値で絶対位置ドライブ開始	MR1値で絶対位置ドライブ開始	MR2値で絶対位置ドライブ開始	MR3値で絶対位置ドライブ開始	説明４
11	ドライブ減速停止				説明５
12	ドライブ即停止				説明５
13	ドライブ速度増加				説明６
14	ドライブ速度減少				説明６
15	タイマー始動				
16	タイマー停止				
17	スプリットパルス開始				説明７
18	スプリットパルス停止				説明７
00	NOP				説明８

説明１： パラメータ値のロード

多目的レジスタMRmの値をそれぞれのパラメータにロードします。

表 2.6-3 パラメータ値のロード

表記	説明
MRm → DV	MRmレジスタの値をドライブ速度(DV)にロードする。
MRm → TP	MRmレジスタの値を移動パルス数(TP)にロードする。
MRm → SP1	MRmレジスタの値をスプリットパルスデータ１(スプリット長とパルス幅)にロードする。
MR0 → LP	MR0レジスタの値を論理位置カウンタ(LP)にロードする。
MR1 → RP	MR1レジスタの値を実位置カウンタ(RP)にロードする。
MR2 → SV	MR2レジスタの値を初速度(SV)にロードする。
MR3 → AC	MR3レジスタの値を加速度(AC)にロードする。

同期動作セットの番号によって、使用されるMRmレジスタは固定されています。
動作コード04hは、同期動作セットの番号によって、MRmレジスタの値をロードするパラメータが変わります。

説明２： パラメータ値のセーブ

それぞれのパラメータの値を多目的レジスタMRmにセーブします。

表 2.6-4 パラメータ値のセーブ

表記	説明
LP → MRm	論理位置カウンタ(LP)の値をMRmレジスタにセーブする。
RP → MRm	実位置カウンタ(RP) の値をMRmレジスタにセーブする。
CT → MRm	現在タイマー値をMRmレジスタにセーブする。
CV → MR0	現在ドライブ速度値をMR0レジスタにセーブする。
CA → MR1	現在加減速度値をMR1レジスタにセーブする。

同期動作セットの番号によって、使用されるMRmレジスタは固定されています。
動作コード08hは、同期動作セット1と2のみ有効で、MRmレジスタに値をセーブするパラメータが変わります。

説明３： 同期パルス信号出力

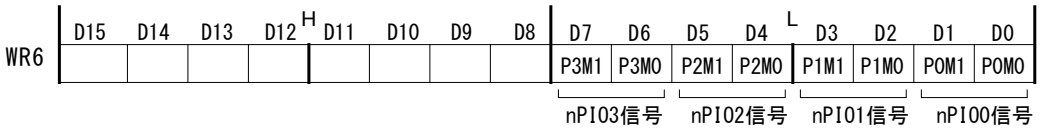
nPIOm (m=0～3)信号から、パルス信号を出力します。
4つの同期動作セット番号に対応するnPIOm信号は固定されています。
この動作を行うためには、次の項目の設定を行う必要があります。

① nPIOm信号の同期パルス出力設定
② 出力パルス信号の論理とパルス幅の設定

同期動作のパルス信号を外部に対して出力するには、汎用入出力信号を同期パルス出力用にモード設定しなければなりません。さらに、この信号をHiパルスで出力するかLowパルスで出力するかの論理設定とパルスの幅を設定する必要があります。これらの設定は、PIO信号設定1命令(21h)およびPIO信号設定2・その他設定命令(22h)で行います。

① nPIOm(m=0～3)信号の同期パルス出力設定

nPIOm信号を同期パルス出力用にモード設定するには、PIO信号設定1命令(21h)で行います。
次の設定を行う必要があります。



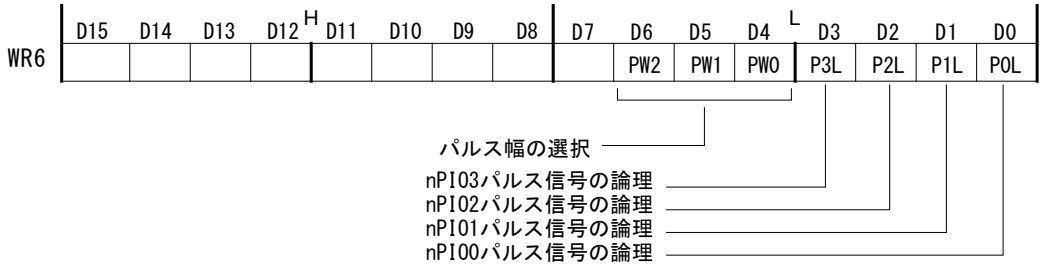
PkM1	PkM0	設定内容
1	1	同期動作出力

k=0～3

使用するnPIOm信号に対応するWR6レジスタの2ビットを同期パルス信号出力用として1,1に設定します。例えばXPIO2信号を使用する場合には、WR6レジスタのD5,D4ビット(P2M1,P2M0)を1,1にセットしてから、X軸指定でPIO信号設定1命令(21h)をWR0レジスタに書き込みます。

② 出力パルス信号の論理とパルス幅の設定

出力パルス信号の論理とパルス幅を設定するには、PIO信号設定2・その他設定(22h)で行います。
次の設定を行う必要があります。



PkL (k=0～3)	パルス信号の論理
0	正論理パルスを出力する。
1	負論理パルスを出力する。

PW2	PW1	PW0	パルス幅 (CLK=16MHz時)
0	0	0	125nsec
0	0	1	312nsec
0	1	0	1μsec
0	1	1	4μsec
1	0	0	16μsec
1	0	1	64μsec
1	1	0	256μsec
1	1	1	1msec

WR6レジスタのD0からD3ビット(P0L～P3L)に使用するnPIOm信号のパルス論理を指定します。0は正論理パルス、1は負論理パルスが出力されます。使用しない信号に対応するビットは0でも1でも構いません。また、WR6レジスタのD4からD6ビット(PW0～PW3)には上表に示すパルス幅を設定します。PIO信号設定2・その他命令(22h)をWR0レジスタに書き込むとWR6レジスタの指定内容が設定されます。

【注意】

- パルス幅の設定は nPIO0～nPIO3 信号すべてに共通です。それぞれの信号ごとにパルス幅を個別に設定することはできません。
- 同期パルス出力の動作が連続して起動がかかる場合、同期パルス出力途中に次の起動がかかると、同期パルスは非アクティブにはならず、起動がかかった時点から再度、指定のパルス幅を出力します。

説明４： MRm値で相対／絶対位置ドライブ起動

ドライブ開始時、MRmレジスタの値が移動パルス数(TP)の値にセットされ、それに応じた相対／絶対位置ドライブが起動されます。

移動パルス数(TP)にMRmレジスタの値が書き込まれるため、この動作(Action)を実行すると移動パルス数(TP)の設定内容が変わります。移動パルス数／終点設定値読み出し命令(46h)で変更された移動パルス数(TP)の値を確認できます。

説明５： ドライブ減速停止／即停止

ドライブを減速停止／即停止させます。

【注意】 補間ドライブを本動作(Action)で停止させた場合、必ず補間軸に対しエラー・終了ステータスクリア命令(79h)を発行してください。

説明６： ドライブ速度増加／減少

ドライブ中の現在ドライブ速度を増加／減少させます。増減値は、あらかじめ速度増減値設定命令(15h)で設定しておく必要があります。

S字ドライブの加減速中や補間ドライブ中では、この動作(Action)は無効となります。

説明７： スプリットパルス開始／停止

スプリットパルス開始は、あらかじめ設定された内容でスプリットパルスを開始します。起動要因発生タイミングによってスプリットパルスの開始ドライブパルスが決定します。スプリットパルス停止は、現在動作中のスプリットパルスを停止させます。起動要因発生タイミングによってスプリットパルスの停止タイミングが決定します。詳細は2.7節を参照してください。

説明８： NOP

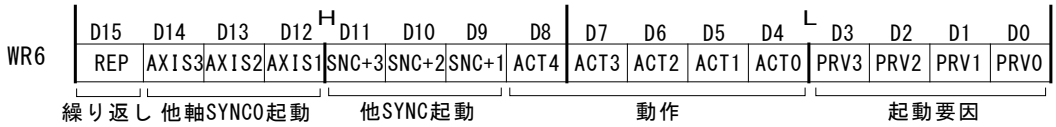
起動要因がアクティブになっても動作(Action)として何も動作をさせないときに指定します。
例えば、ある起動要因で割り込みのみを発生させるときなどに用いることができます。

2.6.3 同期動作の設定

各軸の同期動作の設定として、SYNCm設定、有効設定、無効設定が存在し、これらを設定することで同期動作を行います。

■ SYNCm 設定

4つの同期動作セットを同期動作 SYNCm 設定命令(26h, 27h, 28h, 29h)で設定します。起動要因と動作(Action)、他の同期動作セット起動、同期動作の単一／繰り返しの設定を行います。
WR6 レジスタに設定内容を書き込んだのち、同期動作設定命令を書き込みます。



① 起動要因の設定

起動要因を D3～0 ビット(PRV3～PRV0)の 4 ビットで指定します。
例えば、起動要因に「ドライブ開始」を設定する場合は、コード 3h を指定すればよいので、D3～0 は 0011 となります。
起動要因の詳細については、2.6.1 項を参照してください。

② 動作(Action)の設定

動作(Action)を D8～4 ビット(ACT4～ACT0)の 5 ビットで指定します。
例えば、動作(Action)に「スプリットパルス開始」を設定する場合は、コード 17h を指定すればよいので、D8～4 は 10111 となります。
動作(Action)の詳細については、2.6.2 項(を参照してください。

③ 他の同期動作セット起動の設定

同期動作セットで起動要因がアクティブになったときに、他の同期動作セットの動作(Action)を連動して同時に起動させたい場合にこのビットを設定します。
D11～9 ビット(SNC+3～SNC+1)で指定します。
他の同期動作セットの動作(Action)を起動させる場合は1を、起動させない場合は 0 を設定します。
ビットの指定と起動させる他の同期動作セットの対応は下表の通りです。

表 2.6-5 他の同期動作セット起動

自同期動作セット	D11(SNC+3)	D10(SNC+2)	D9(SNC+1)
SYNC0	SYNC3 起動	SYNC2 起動	SYNC1 起動
SYNC1	SYNC0 起動	SYNC3 起動	SYNC2 起動
SYNC2	SYNC1 起動	SYNC0 起動	SYNC3 起動
SYNC3	SYNC2 起動	SYNC1 起動	SYNC0 起動

この機能を用いることで、一つの起動要因に対して、複数の動作(Action)を同時に起動させることができるため、より複雑な同期動作を行わせることができます。
設定は、例えば自身の同期動作セットを SYNC0 とします。このとき、SYNC0 の起動要因がアクティブになったとき、SYNC1、2 の動作(Aciton)を起動させたい場合、上記の表をもとに、D9 および D10 ビットを1にします。この設定により、SYNC0 の起動要因がアクティブになったとき、SYNC0 の動作(Action)以外に、SYNC1,2 の動作(Action)も同時に起動されます。このとき、SYNC1,2 は起動要因は NOP にしておき動作(Action)のみを設定します。さらに同期動作有効設定命令で有効状態にしておく必要があります。

④ 他軸の同期動作セット 0 起動の設定

同期動作セットで起動要因がアクティブになったときに、他軸の同期動作セット 0(SYNC0)の動作(Action)を連動して同時に起動させたい場合にこのビットを設定します。
D14～12 ビット(AXIS3～AXIS1)で指定します。
他軸の同期動作セット 0(SYNC0)の動作(Action)を起動させる場合は1を、起動させない場合は 0 を設定します。
ビットの指定と起動させる他軸の同期動作セット 0(SYNC0)の対応は下表の通りです。

表 2.6-6 他軸の同期動作セット 0(SYNC0)起動

自軸	D14(AXIS3)	D13(AXIS2)	D12(AXIS1)
X	U 軸 SYNC0 起動	Z 軸 SYNC0 起動	Y 軸 SYNC0 起動
Y	X 軸 SYNC0 起動	U 軸 SYNC0 起動	Z 軸 SYNC0 起動
Z	Y 軸 SYNC0 起動	X 軸 SYNC0 起動	U 軸 SYNC0 起動
U	Z 軸 SYNC0 起動	Y 軸 SYNC0 起動	X 軸 SYNC0 起動

この機能を用いることで、一つの起動要因に対して、他軸の動作(Action)を同時に起動させることができるため、より複雑な同期動作を行わせることができます。
設定は、例えば X 軸の同期動作セットとして SYNC0 を設定します。このとき、X 軸の SYNC0 の起動要因がアクティブになったとき、Y、Z 軸の SYNC0 の動作(Aciton)を起動させたい場合、上の表をもとに、D12 および D13ビットを1にします。この設定により、X 軸の SYNC0 の起動要因がアクティブになったとき、X 軸の SYNC0 の動作(Action)以外に、Y、Z 軸の SYNC0 の動作(Action)も同時に起動されます。このとき、Y、Z 軸の SYNC0 の起動要因は NOP にしておき動作(Action)のみを設定します。さらに同期動作有効設定命令で有効状態にしておく必要があります。

⑤ 同期動作セットの繰り返し設定

同期動作セットの有効状態を、同期動作が一度起動したのち無効状態にするか否かを設定することができます。
繰り返しを有効にする場合は D15 ビット(REP)に 1 を、1 回のみ有効にする場合は 0 を設定します。
繰り返しを有効にした場合、起動要因がアクティブになるたびに繰り返し同期動作が起動します。1回のみ有効にした場合、起動要因がアクティブになった初めの 1 回だけ同期動作が起動します。

【注意】繰り返し設定有効時、起動要因をドライブ停止、動作(Action)を相対位置ドライブ起動とした場合、ドライブ終了→ドライブ開始の動作がエンドレスで続いてしまいます。停止命令では止まりません。同期動作無効設定命令で止めます。

■ 有効設定

同期動作有効設定命令(81h～8Fh)で各同期動作セットを有効状態にします。
同期動作セットが有効状態のとき、起動要因がアクティブになると動作(Aciton)が起動します。

4つの同期動作セットにそれぞれ対応した命令コードがあり、同期動作セット SYNC0 は 81h、同期動作セット SYNC1 は 82h、同期動作セット SYNC2 は 84h、同期動作セット SYNC3 は 88h、となります。
これらの命令は組み合わせで複数同時に有効にすることが可能です。例えば、83h 命令を実行すると SYNC0,1 が有効状態になります。命令コードの組み合わせは、表 2.6-7 を参照してください。

SYNCm 設定で REP=0 に設定してある場合、一度、同期動作が実行されると、その同期動作セットは無効状態になり、再度起動要因がアクティブになっても同期動作は起動しません。REP=1にすると、同期動作が実行されても、その同期動作セットは有効状態のままです。
同期動作実行により、一度無効状態になった同期動作セットを再び有効状態にするには、再度、同期動作有効設定命令を発行する必要があります。

PIO 信号設定2・その他命令(22h)で、ERRDE=1 にすると、エラー発生時(RR0 レジスタのエラービットが1になった状態)に、全ての同期動作セットは無効状態に変わります。この状態になると、エラー状態をクリアしない限り、同期動作有効設定命令を発行しても、有効状態になりません。エラー状態をクリアするには、エラー・終了ステータスクリア命令(79h)を発行します。

4つの同期動作セットの有効／無効状態は、RR3 レジスタ ページ1の D3～D0 ビット(SYNC3～SYNC0)で確認することができます。

■ 無効設定

同期動作無効設定命令(91h～9Fh)で各同期動作セットを無効状態にします。
同期動作セットが無効状態のとき、起動要因がアクティブになっても動作(Aciton)は起動しません。
リセット時には、4つの同期動作セットは全て無効状態です。

4つの同期動作セットにそれぞれ対応した命令コードがあり、同期動作セット SYNC0 は 91h、同期動作セット SYNC1 は 92h、同期動作セット SYNC2 は 94h、同期動作セット SYNC3 は 98h、となります。
同期動作有効設定命令同様、組み合わせで複数同時に無効にすることが可能です。命令コードの組み合わせは、表 2.6-7 を参照してください。

同期動作が無効状態に変化するのは、「同期動作無効命令発行時」、「PIO 信号設定2・その他設定命令(22h)でエラー発生時に同期動作無効設定(D7:ERRDE)を有効に設定をしたときに、エラーが発生時」、「同期動作 1 回(繰り返し無効)設定時に、同期動作起動後」の3つです。

4つの同期動作セットの有効／無効状態は、RR3 レジスタ ページ1の D3～D0 ビット(SYNC3～SYNC0)で確認することができます。

表 2.6-7 同期動作有効，無効，起動命令コードと対応する同期動作セット

命令コード(Hex)			同期動作セット			
有効設定	無効設定	起動	同期動作セット３ SYNC3	同期動作セット２ SYNC2	同期動作セット１ SYNC1	同期動作セット０ SYNC0
81	91	A1	—	—	—	○
82	92	A2	—	—	○	—
83	93	A3	—	—	○	○
84	94	A4	—	○	—	—
85	95	A5	—	○	—	○
86	96	A6	—	○	○	—
87	97	A7	—	○	○	○
88	98	A8	○	—	—	—
89	99	A9	○	—	—	○
8A	9A	AA	○	—	○	—
8B	9B	AB	○	—	○	○
8C	9C	AC	○	○	—	—
8D	9D	AD	○	○	—	○
8E	9E	AE	○	○	○	—
8F	9F	AF	○	○	○	○

○：有効設定命令実行時は有効状態、無効設定命令実行時は無効状態になり、起動命令実行時は起動します。
—：有効、無効設定命令実行時でも、共に状態に変化はありません。起動命令実行時は起動しません。

2.6.4 同期動作の実行

■ 同期動作の実行手順

次の手順で同期動作を行います。

① **同期動作 SYNCm 設定命令(26h～29h)により、起動要因と動作(Action)を設定する。**
② **同期動作有効設定命令(81h～8Fh)により、同期動作セットを有効にする。**
③ **設定した起動要因が発生すると、同期動作が作動する。**

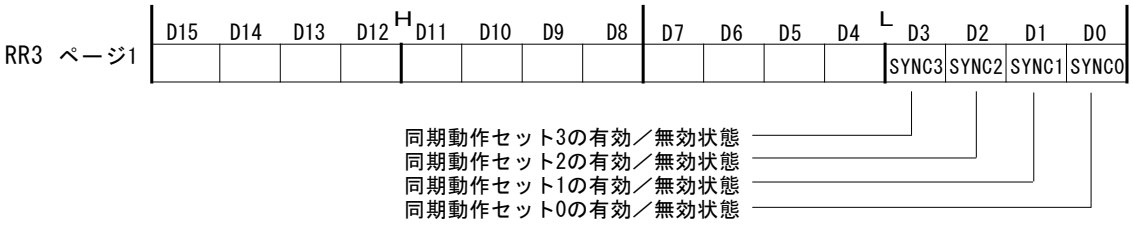
■ 同期動作の起動命令による起動

同期動作は命令によって起動することもできます。同期動作起動命令(A1h～Ah)によって起動します。命令コードによって複数の同期動作セットを同時に起動することができます。命令コードと起動する同期動作 SYNC3～0 の対応は表 2.6-7 を参照してください。

同期動作起動命令によって同期動作を起動するためには、同期動作有効設定命令で指定の同期動作セットを有効にしておく必要があります。

■ 同期動作有効／無効状態

RR3 レジスタ ページ 1 の D3～D0 ビット(SYNC3～SYNC0)で同期動作セットの状態を確認することができます。
ビットが1のときは同期動作セットが有効状態、0 のときは無効状態となります。


RR3 ページ1
D15 D14 D13 D12 H D11 D10 D9 D8 D7 D6 D5 D4 L D3 D2 D1 D0
SYNC3 SYNC2 SYNC1 SYNC0
同期動作セット3の有効／無効状態
同期動作セット2の有効／無効状態
同期動作セット1の有効／無効状態
同期動作セット0の有効／無効状態

2.6.5 同期動作による割り込み発生

同期動作起動時に割り込みを発生させる事ができます。
WR1 レジスタの D15～D12 ビット(SYNC3～SYNC0)に設定します。
これらのビットを1にすると、ビットに対応した同期動作セットの起動要因がアクティブになると割り込みが発生します。
割り込み機能に関しては、2.10 節を参照してください。

2.6.6 同期動作の実例

■ 例 1 X 軸ドライブ中に指定位置 15000 を通過したら同期パルスを XPIO0 に出力する

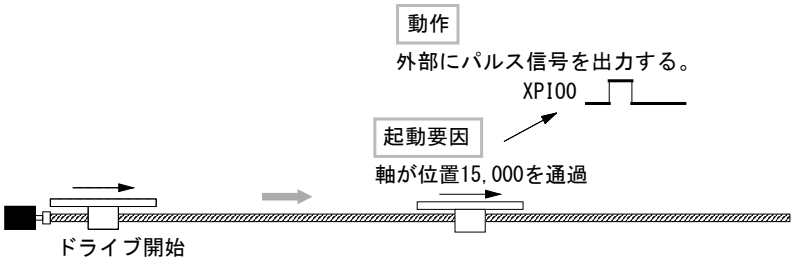


図 2.6-8 同期動作の実例 1

【プログラム例】

```
// ドライブの設定(1000 PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                  // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ                  // ドライブ速度  1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                  // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

// MR0 の設定
WR6 ← 3A98h ﾗｲﾄ                  // MR0  15000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0110h ﾗｲﾄ

// 多目的レジスタモード設定
WR6 ← 0000h ﾗｲﾄ                  // D1,D0      00 MOT1,0 : MR0 の比較対象  論理位置カウンタ
                                  // D3,D2      00 MOC1,0 : MR0 の比較条件  ≧
WR0 ← 0120h ﾗｲﾄ

// PIO 信号設定 1
WR6 ← 0003h ﾗｲﾄ                  // D1,D0      11 POM1,0 : PIO0 信号  同期動作出力
WR0 ← 0121h ﾗｲﾄ

// PIO 信号設定 2
WR6 ← 0070h ﾗｲﾄ                  // D0         0 POL : PIO0 パルス信号の論理  正論理
                                  // D6～D4     111 PW2～0 : パルス幅  1msec(クロック 16MHz 時)
WR0 ← 0122h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0091h ﾗｲﾄ                  // D3～D0     0001 PREV3～0  : 同期動作の起動要因  MR0 比較が真に変化した
                                  // D8～D4     01001 ACT4～0 : 同期動作の動作  同期パルス出力
WR0 ← 0126h ﾗｲﾄ

// SYNC0 有効
WR0 ← 0181h ﾗｲﾄ


// ドライブ開始
WR0 ← 0152h ﾗｲﾄ                  // ＋方向連続パルスドライブ開始
```

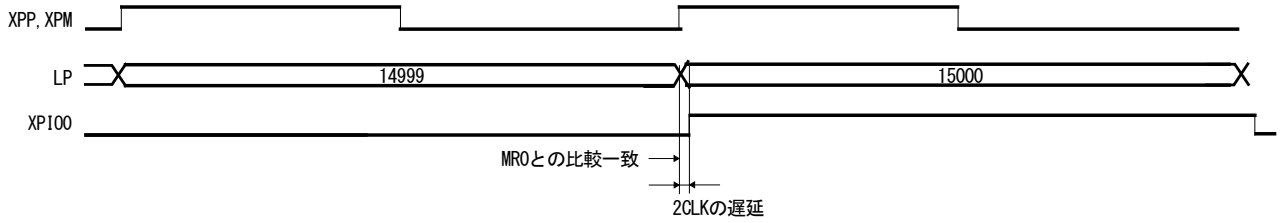


図 2.6-9 同期動作の実例１のタイミング

2.6.7 項から、起動要因発生の遅延時間は1CLK、動作(Action)の遅延時間は1CLK となるので、この同期動作での遅延時間は2CLK(125nsec)となります。

■ 例 2 X 軸ドライブ中に外部信号が入力したら位置データをセーブする

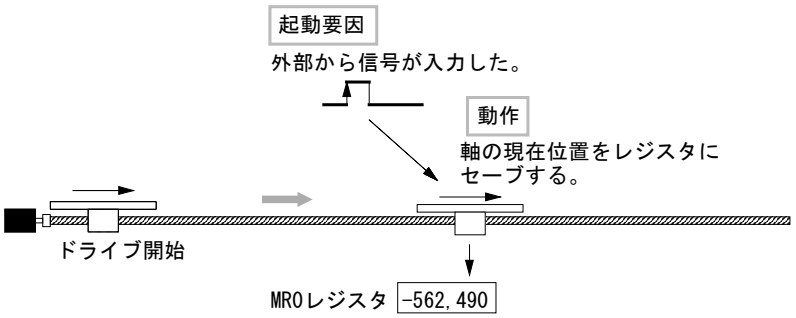


図 2.6-10 同期動作の実例２

【プログラム例】

```
// ドライブの設定(1000 PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                  // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ                  // ドライブ速度  1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                  // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

// PIO 信号設定 1
WR6 ← 0000h ﾗｲﾄ                  // D1,D0      00 POM1,0 : PIO0 信号  汎用・同期入力
WR0 ← 0121h ﾗｲﾄ

// 割り込みの設定
WR0 ← 011Fh ﾗｲﾄ                  // X軸選択
WR1 ← 1000h ﾗｲﾄ                  // D12        1 SYNC0 : 同期動作 SYNC0 起動時

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 005Ah ﾗｲﾄ                  // D3～D0     1010 PREV3～0   : 同期動作の起動要因  XPIOm 入力↑
                                  // D8～D4     00101 ACT4～0   : 同期動作の動作  セーブ LP → MRm
WR0 ← 0126h ﾗｲﾄ

// SYNC0 有効
WR0 ← 0181h ﾗｲﾄ

// ドライブ開始
WR0 ← 0152h ﾗｲﾄ                  // ＋方向連続パルスドライブ開始
```

↓

SYNC0が起動し割り込み発生

↓

```
// MR0 にセーブされた論理位置カウンタ値を読み出し
WR0 ← 0134h ﾗｲﾄ
RR6 → リード
RR7 → リード
```

2.6.7 項から、起動要因発生の遅延時間は最小で 0、最大で1CLK、動作(Action)の遅延時間は1CLK となるので、この同期動作での遅延時間は最小1CLK(62.5nsec)、最大2CLK(125nsec)となります。

■ 例 3 X 軸ドライブ中に、指定位置 A(10000)から指定位置 B(55000)までを通過する時間を求める

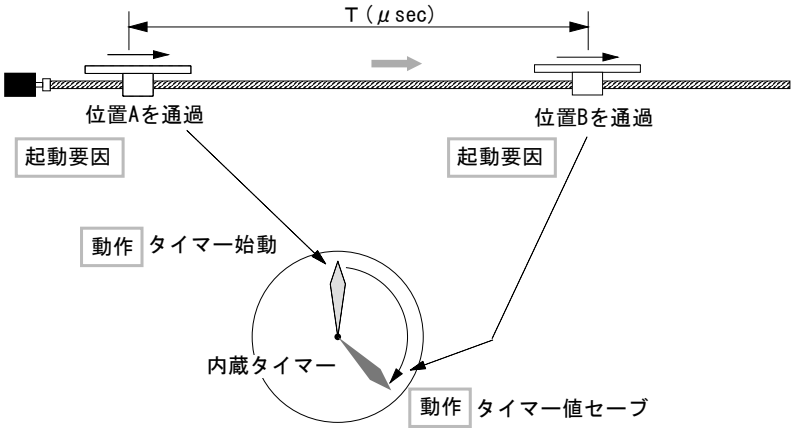
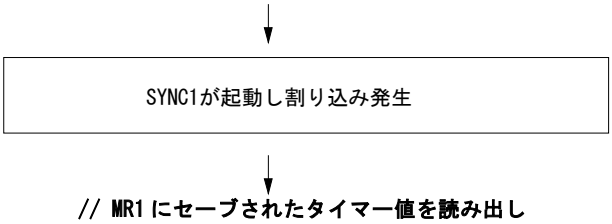


図 2.6-11 同期動作の実例３

【プログラム例】

```
// ドライブの設定(10K PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                  // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ                  // ドライブ速度  10K PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                  // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

// 指定位置を MRm レジスタに設定
// MR0 の設定(指定位置 A : 10000)
WR6 ← 2710h ﾗｲﾄ                  // MR0  10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0110h ﾗｲﾄ

// MR1 の設定(指定位置 B : 55000)
WR6 ← D6D8h ﾗｲﾄ                  // MR1  55000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0110h ﾗｲﾄ

// タイマー値の設定
WR6 ← FFFFh ﾗｲﾄ                  // タイマー値  2147483647(最大値)
WR7 ← 7FFFh ﾗｲﾄ
WR0 ← 0116h ﾗｲﾄ

// 割り込みの設定
WR0 ← 011Fh ﾗｲﾄ                  // X軸選択
WR1 ← 2000h ﾗｲﾄ                  // D13        1 SYNC1 : 同期動作 SYNC1 起動時

// 多目的レジスタモード設定
WR6 ← 0000h ﾗｲﾄ                  // D1,D0      00 MOT1,0 : MR0 の比較対象  論理位置カウンタ
                                  // D3,D2      00 MOC1,0 : MR0 の比較条件  ≧
                                  // D5,D4      00 M1T1,0 : MR1 の比較対象  論理位置カウンタ
                                  // D7,D6      00 M1C1,0 : MR1 の比較条件  ≧
WR0 ← 0120h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0151h ﾗｲﾄ                  // D3～D0     0001 PREV3～0  : 同期動作の起動要因  MRm 比較が真に変化した
                                  // D8～D4     10101 ACT4～0  : 同期動作の動作  タイマー始動
WR0 ← 0126h ﾗｲﾄ

// 同期動作 SYNC1 設定
WR6 ← 0071h ﾗｲﾄ                  // D3～D0     0001 PREV3～0  : 同期動作の起動要因  MRm 比較が真に変化した
                                  // D8～D4     00111 ACT4～0  : 同期動作の動作  セーブ CT → MRm
WR0 ← 0127h ﾗｲﾄ

// SYNC0,1 有効
WR0 ← 0183h ﾗｲﾄ
```

// ドライブ開始
WR0 ← 0152h ﾗｲﾄ　　// ＋方向連続パルスドライブ開始



// MR1 にセーブされたタイマー値を読み出し
WR0 ← 0135h ﾗｲﾄ
RR6 → リード
RR7 → リード

// タイマー停止
WR0 ← 0174h ﾗｲﾄ

2.6.7 同期動作の遅延時間

同期動作の遅延時間は、下表に示す起動要因発生からの遅延と、動作(Action)までの遅延の合計で求められます。

■ 起動要因発生からの遅延 1CLK=62.5nsec (CLK=16MHz の時)

表 2.6-8 起動要因発生からの遅延

起動要因		遅延開始の定義	遅延時間(CLK)		
			最小	標準	最大
MRm 比較が真に変化した	論理位置カウンタ	論理位置カウンタが MRm 値との比較条件に一致する時のドライブパルスの↑から		1	
	実位置カウンタ	実位置カウンタが MRm 値との比較条件に一致する時の nECA/B 入力信号の↑↓から	2		3
	現在速度	現在速度が MRm 値との比較条件に一致した時から		1	
	現在タイマー値	現在タイマー値が MRm 値との比較条件に一致した時から		1	
タイムアップ		現在タイマー値が指定した値になった時から		0	
ドライブ開始		ドライブ命令書き込み時の WRN 信号の↑から	2		3
加減速ドライブの定速域が開始		CNST 信号↑から		0	
加減速ドライブの定速域が終了		CNST 信号↓から		0	
ドライブ終了		最終ドライブパルスの Low レベル終了から		1	
スプリットパルス開始		第 1 nSPLTP 信号↑から(開始パルス有効時)		0	
スプリットパルス終了		最終 nSPLTP 信号↓から(正論理のとき)		2	
スプリットパルス出力		nSPLTP 信号↑から(正論理のとき)		0	
nPIOm 入力↑		nPIOm 信号の↑から (内蔵フィルタ無効時)	0		1
nPIOm 入力↓		nPIOm 信号の↓から (内蔵フィルタ無効時)	0		1
nPIOm 入力 Low かつ nPIO(m+4)↑		nPIO(m+4)信号の↑から (内蔵フィルタ無効時)	0		1
nPIOm 入力 Hi かつ nPIO(m+4)↑		nPIO(m+4)信号の↑から (内蔵フィルタ無効時)	0		1
nPIOm 入力 Low かつ nPIO(m+4)↓		nPIO(m+4)信号の↓から (内蔵フィルタ無効時)	0		1
nPIOm 入力 Hi かつ nPIO(m+4)↓		nPIO(m+4)信号の↓から (内蔵フィルタ無効時)	0		1
起動命令		同期動作起動命令書き込み時の WRN 信号の↑から	1		2

■ 動作(Action)までの遅延 1CLK=62.5nsec (CLK=16MHz の時)

表 2.6-9 動作(Action)までの遅延

動作	遅延終了の定義	遅延時間(CLK)
ロード MRm → DV	MRm の値が DV にロードされるまで	1
ロード MRm → TP	MRm の値が TP にロードされるまで	1
ロード MRm → SP1	MRm の値が SP1 にロードされるまで	1
ロード MRm → LP(SYNC0), RP(SYNC1), SV(SYNC2), AC(SYNC3)	MRm の値が LP(SYNC0), RP(SYNC1), SV(SYNC2), AC(SYNC3)にロードされるまで	1
セーブ LP → MRm	LP の値が MRm にセーブされるまで	1
セーブ RP → MRm	RP の値が MRm にセーブされるまで	1
セーブ CT → MRm	CT の値が MRm にセーブされるまで	1
セーブ CV(SYNC0), CA(SYNC1) → MRm	CV(SYNC0), CA(SYNC1)の値が MRm にセーブされるまで	1
同期パルス nPIOm 出力	同期パルス nPIOm 信号の↑まで	1
相対位置ドライブ起動	第 1 ドライブパルスの↑まで	3
反相対位置ドライブ起動	第 1 ドライブパルスの↑まで	3
絶対位置ドライブ起動	第 1 ドライブパルスの↑まで	3
＋方向連続パルスドライブ起動	第 1 ドライブパルスの↑まで	3
－方向連続パルスドライブ起動	第 1 ドライブパルスの↑まで	3
MRm 値の移動パルス数で相対位置ドライブ	第 1 ドライブパルスの↑まで	4
MRm 値の終点へ絶対位置ドライブ	第 1 ドライブパルスの↑まで	4
ドライブ減速停止	減速を開始するまで	(※1)
ドライブ即停止	ドライブを停止するまで	(※1)
ドライブ速度増加	変更された速度に向かって速度増加を開始するまで	1
ドライブ速度減少	変更された速度に向かって速度減少を開始するまで	1
タイマー始動	タイマーが開始されるまで	1
タイマー停止	タイマーが停止されるまで	1
スプリットパルス開始	nSPLTP 信号↑まで(開始パルス有りのとき)	(※2)
スプリットパルス停止	nSPLTP 信号↓まで	(※3)
割り込み	INTON 信号の↓まで	1

(※1)現在出力中の1ドライブパルスが終了するまでの時間
(※2)スプリットパルスはドライブパルスに同期しているため、最大で1ドライブパルス周期の遅延となります。
(※3)現在出力中のスプリットパルスが終了するまでの時間

■ 遅延計算例

例えば、起動要因「nPIOm 入力↑」から動作(Action)「セーブ LP → MRm」までの遅延時間は、起動要因「nPIOm 入力↑」遅延時間(0～1CLK)と動作(Action)「セーブ LP → MRm」遅延時間(1CLK)を合計して、最小 1CLK から最大 2CLK となります。CLK=16MHz 時には、最小 62.5nsec から最大 125nsec となります。

■ 他 SYNC 起動の遅延

他SYNC起動を実行した場合、自同期動作セットの動作(Action)起動に比べ、1CLK遅れて動作(Action)が起動されます。

■ 他軸の SYNC0 起動の遅延

他軸SYNC0起動を実行した場合、自同期動作セットの動作(Action)起動に比べ、1CLK遅れて動作(Action)が起動されます。

2.7 スプリットパルス

各軸のドライブ中にドライブパルスと同期したスプリットパルスを出力する機能です。
モータの回転や軸移動に同期させて、一定のパルス間隔で別の仕事をさせたいときに便利な機能です。
補間ドライブ時も出力させる事ができます。
スプリットパルスのパルス幅、スプリット長(周期)、スプリットパルス数を設定することができます。また、パルスの論理レベル、開始パルス有り／無しを指定することができます。
スプリットパルスは表 2.7-1 に示す端子から出力されます。

表 2.7-1 各軸のスプリットパルス端子

軸	信号名	端子番号
X	XSPLTP	65
Y	YSPLTP	84
Z	ZSPLTP	103
U	USPLTP	122

ドライブ中のスプリットパルスの開始は命令、あるいは同期動作によって行います。同期動作を用いると、指定の位置カウンタの値から開始させたり、外部信号の↑から開始させたりすることができます。

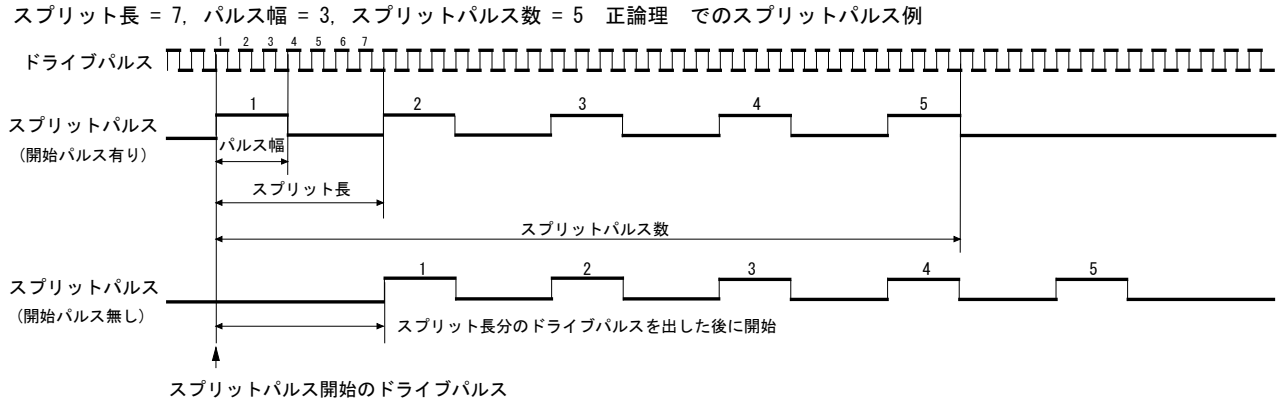


図 2.7-1 スプリットパルス 例

2.7.1 スプリットパルスの設定

スプリットパルスを行わせるために、次のパラメータとモード設定が必要です。

■ スプリット長、パルス幅の設定
スプリット長、パルス幅の設定は、スプリットパルス設定1命令(17h)によって行います。WR6 レジスタにはスプリット長、WR7レジスタにはパルス幅を設定します。スプリット長とパルス幅の設定単位はドライブパルスの数です。
スプリットパルスの機能上、スプリット長＞パルス幅の設定をしてください。
スプリット長は 2～65535、パルス幅は 1～65534 の範囲で設定が可能です。
設定された内容は、スプリットパルス設定1 読み出し命令(47h)で確認する事ができます。

スプリット長(周期)、パルス幅はスプリットパルス動作中でも設定内容を変更することが可能です。

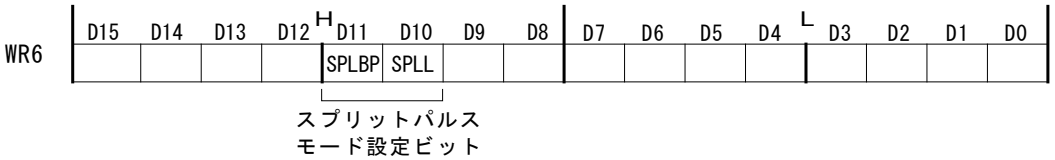
■ スプリットパルス数の設定
スプリットパルス数の設定は、スプリットパルス設定 2 命令(18h)によって行います。WR6 レジスタに設定します。
0～65535 まで設定が可能です。0 を指定すると無限となります。開始後は、スプリットパルス停止命令あるいはドライブが停止されるまで、スプリットパルスを出力し続けます。

スプリットパルス数は、スプリットパルス動作中でも設定内容を変更することが可能です。

■ スプリットパルスのモードの設定

スプリットパルスの動作モードをPIO信号設定2・その他設定命令(22h)で設定します。

スプリットパルス開始時の開始パルス有り無し、およびパルスの出力論理を、WR6 レジスタの D10、D11 ビットに設定します。



スプリットパルスのパルス論理を D10 ビット(SPLL)で設定します。
下図のように、0 を指定すると正論理、1 を指定すると負論理となります。

図 2.7-2 スプリットパルスのパルス論理

スプリットパルスの開始パルスの有り無しを D11 ビット(SPLBP)で設定します。
D11 ビット(SPLBP)に1を指定すると開始パルス有り、0 を設定すると開始パルス無しになります。
開始パルス有りを設定した場合、スプリットパルス開始後、次のドライブパルスからスプリットパルスを出力します。開始パルス無しの場合は、スプリットパルス開始後、スプリット長分のドライブパルス数を出力し終えた後、最初のスプリットパルスを出力します。

2.7.2 スプリットパルスの開始／停止

■ スプリットパルスの開始

スプリットパルス開始命令(75h)および同期動作によって開始します。
命令が書き込まれたとき、または同期動作の動作（Action）が開始されるときの次のドライブパルスがスプリットパルスの開始ドライブパルスとなります。

■ スプリットパルスの停止

スプリットパルス出力は、以下の 3 つの要因のうち、いずれかによって停止します。

- 指定したスプリットパルス数を出力し終えたとき
- スプリットパルス停止命令、または同期動作の動作（Action）として停止がかかったとき
- ドライブが停止したとき

指定したスプリットパルス数を出力し終えて停止する場合、指定したスプリットパルス数の最後のスプリットパルスが OFF 状態になったときに停止します。

スプリットパルス停止命令(76h)や同期動作によってスプリットパルスを停止する場合、スプリットパルスが ON 状態の時は、そのスプリットパルス幅を出力し終えてから停止します。停止を実行したとき、スプリットパルスが OFF 状態の時は、スプリットパルス停止命令および同期動作実行のタイミングで停止します。

ドライブ停止によってスプリットパルス出力が停止する場合、スプリットパルスの出力状態に関係なく、ドライブ停止のタイミングでスプリットパルスは OFF 状態になり、停止します。

■ スプリットパルス動作確認

RR3 レジスタ ページ1の D11 ビット(SPLIT)でスプリットパルス動作中を確認する事ができます。
D11ビット(SPLIT)が1のときはスプリットパルスが動作中、0 のときはスプリットパルスは停止中となります。

2.7.3 同期動作におけるスプリットパルス

スプリットパルスの動作を同期動作で用いる事ができます。
同期動作の起動要因として、「スプリットパルス開始時」、「スプリットパルス出力時」、「スプリットパルス終了時」の3種類を指定できます。
同期動作の動作(Action)として、「スプリットパルスの開始」、「スプリットパルスの停止」、「多目的レジスタのデータをスプリットパルスデータ(スプリット長、パルス幅)にロード」、の3種類を指定できます。
これらの機能の詳細は、2.6 節を参照してください。

2.7.4 スプリットパルスによる割り込み発生

スプリットパルス動作に関する割り込みを発生させる事ができます。
WR1 レジスタの D10 および D11 ビットに設定します。
D10 ビット(SPLTP)を1にすると、スプリットパルスごとのパルスの↑で割り込みが発生します。(スプリットパルス正論理設定時)
D11 ビット(SPLTE)を1にすると、スプリットパルスの動作が終了したときに割り込みが発生します。
割り込み機能に関しては、2.10 節を参照してください。

2.7.5 スプリットパルスの注意点

- スプリットパルス開始パルス有りの場合、第一パルスのみ出力タイミングが異なります。詳細は 11.7 節を参照してください。
- スプリットパルス動作中、指定のスプリットパルス数を出し終える前に停止命令などで停止をした場合、停止後に再度、スプリットパルスを開始すると、スプリットパルス数のカウントは1から開始されます。

2.7.6 スプリットパルスの実例

■ 例1 X 軸ドライブ開始からスプリットパルスを開始

スプリットパルス開始命令発行後、ドライブを開始させ、ドライブと同時にスプリットパルスを出力させる例です。

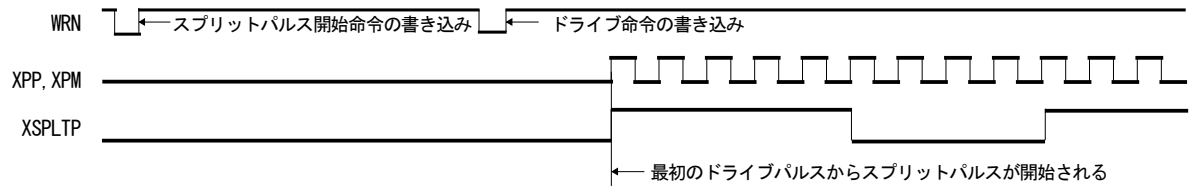


図 2.7-3 ドライブ開始によるスプリットパルス出力タイミング

【プログラム例】

```
// ドライブの設定(1000 PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                    // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ                    // ドライブ速度  1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                    // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

// スプリットパルスの設定
// スプリット長、パルス幅の設定
WR6 ← 0009h ﾗｲﾄ                    // スプリット長  9
WR7 ← 0005h ﾗｲﾄ                    // パルス幅  5
WR0 ← 0117h ﾗｲﾄ

// スプリットパルス数の設定
WR6 ← 000Ah ﾗｲﾄ                    // スプリットパルス数  10
WR0 ← 0118h ﾗｲﾄ

// スプリットパルス論理、開始パルスの設定
WR6 ← 0800h ﾗｲﾄ                    // D10        0 SPLL : パルス論理  正
                                    // D11        1 SPLBP : 開始パルス  有
WR0 ← 0122h ﾗｲﾄ

// スプリット開始(ドライブ開始前にスプリットパルス開始命令を発行)
WR0 ← 0175h ﾗｲﾄ

// ドライブ開始
WR0 ← 0152h ﾗｲﾄ                    // ＋方向連続パルスドライブ開始
```

ドライブ開始後、最初のドライブパルスがスプリットパルスの開始ドライブパルスとなります。
スプリットパルス開始命令発行後、ドライブが開始されるまでスプリットパルスは出力されませんが、RR3 レジスタ ページ1の D11 ビット(SPLIT)はスプリットパルス開始命令発行のタイミングで1になります。

■ 例 2 X 軸の指定位置 5000 からスプリットパルスを開始

ドライブ開始後、論理位置が 5000 になった時点からスプリットパルスを開始させる例です。同期動作の機能を用いて行います。

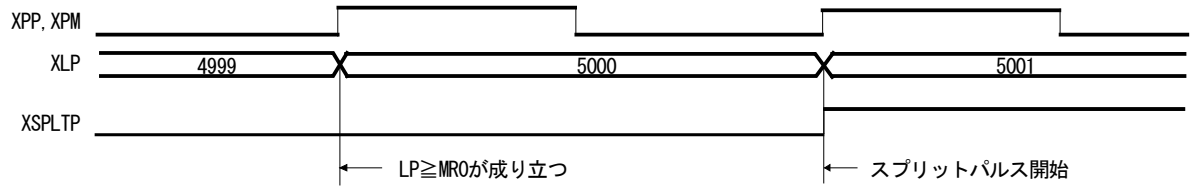


図 2.7-4 MRm 比較によるスプリットパルス出力タイミング

【プログラム例】

```
// ドライブの設定(1000 PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                  // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ                  // ドライブ速度  1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                  // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

// スプリットパルスの設定
// スプリット長、パルス幅の設定
WR6 ← 0008h ﾗｲﾄ                  // スプリット長  8
WR7 ← 0005h ﾗｲﾄ                  // パルス幅  5
WR0 ← 0117h ﾗｲﾄ

// スプリットパルス数の設定
WR6 ← 000Ah ﾗｲﾄ                  // スプリットパルス数  10
WR0 ← 0118h ﾗｲﾄ

// スプリットパルス論理、開始パルスの設定
WR6 ← 0800h ﾗｲﾄ                  // D10        0 SPLL : パルス論理  正
                                  // D11        1 SPLBP : 開始パルス  有
WR0 ← 0122h ﾗｲﾄ

// 多目的レジスタの設定
// MR0 の設定
WR6 ← 1388h ﾗｲﾄ                  // MR0  5000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0110h ﾗｲﾄ

// 多目的レジスタモード設定
WR6 ← 0000h ﾗｲﾄ                  // D1,D0      00 MOT1,0 : MR0 の比較対象  論理位置カウンタ
                                  // D3,D2      00 MOC1,0 : MR0 の比較条件  ≧
WR0 ← 0120h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0171h ﾗｲﾄ                  // D3～D0     0001 PREV3～0  : 同期動作の起動要因  MRm 比較が真に変化した
                                  // D8～D4     10111 ACT4～0  : 同期動作の動作  スプリットパルス開始
WR0 ← 0126h ﾗｲﾄ

// SYNC0 有効
WR0 ← 0181h ﾗｲﾄ

// ドライブ開始
WR0 ← 0152h ﾗｲﾄ                  // ＋方向連続パルスドライブ開始
```

比較値を 5000、比較条件を≧とした場合、スプリットパルスが開始される論理位置カウンタの値は図のように 5001 となります。つまり、比較条件が真になった次のドライブパルスがスプリットパルスの開始ドライブパルスになります。

■ 例 3　X 軸 S 字加減速ドライブの定速区間中、スプリットパルスを出力

S 字加減速ドライブの定速区間中、スプリットパルスを出力する例です。同期動作の機能を用いて行います。

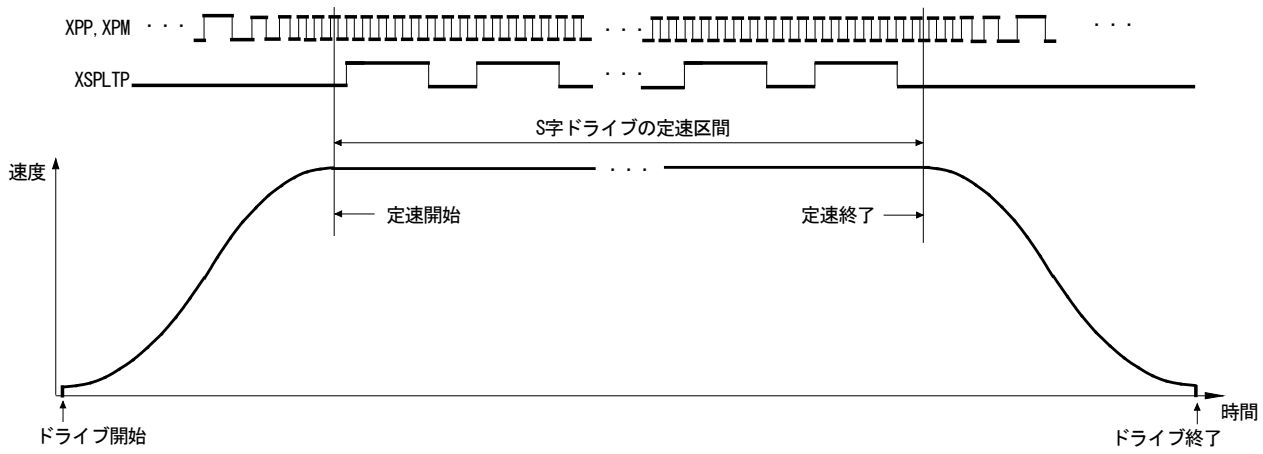


図 2.7-5　S 字加減速ドライブの定速区間にスプリットパルスを出力

【プログラム例】

```
// S 字加減速ドライブの設定
WR6 ← 000Ah ﾗｲﾄ                    // 初速度  10 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 0FA0h ﾗｲﾄ                    // ドライブ速度  4000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← FFFFh ﾗｲﾄ                    // 加速度  536870911(最大値)
WR7 ← 1FFFh ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

WR6 ← A048h ﾗｲﾄ                    // 加速度増加率  893K PPS/SEC2
WR7 ← 000Dh ﾗｲﾄ
WR0 ← 0100h ﾗｲﾄ

WR6 ← 9C40h ﾗｲﾄ                    // 移動パルス数  40000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                    // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

WR3 ← 0104h ﾗｲﾄ                    // D3          1 SACC : S 字加減速

// スプリットパルスの設定
// スプリット長、パルス幅の設定
WR6 ← 0008h ﾗｲﾄ                    // スプリット長  8
WR7 ← 0005h ﾗｲﾄ                    // パルス幅  5
WR0 ← 0117h ﾗｲﾄ

// スプリットパルス数の設定
WR6 ← 0000h ﾗｲﾄ                    // スプリットパルス数  無限
WR0 ← 0118h ﾗｲﾄ

// スプリットパルス論理、開始パルスの設定
WR6 ← 0800h ﾗｲﾄ                    // D10         0 SPLL : パルス論理  正
                                   // D11         1 SPLBP : 開始パルス  有
WR0 ← 0122h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0174h ﾗｲﾄ                    // D3～D0      0100 PREV3～0  : 同期動作の起動要因  ドライブ定速域開始
                                   // D8～D4      10111 ACT4～0  : 同期動作の動作  スプリットパルス開始
                                   // D15         0 REP          : 同期動作の繰り返し  必ず無効に設定
WR0 ← 0126h ﾗｲﾄ
```

```
// 同期動作 SYNC1 設定
WR6 ← 0185h ﾗｲﾄ                  // D3～D0     0101 PREV3～0   : 同期動作の起動要因　ドライブ定速域終了
                                  // D8～D4     11000 ACT4～0   : 同期動作の動作　スプリットパルス停止
                                  // D15        0 REP           : 同期動作の繰り返し　必ず無効に設定
WR0 ← 0127h ﾗｲﾄ

// SYNC0,1 有効
WR0 ← 0183h ﾗｲﾄ

// ドライブ開始
WR0 ← 0150h ﾗｲﾄ                  // 相対位置ドライブ開始
```

■ 例 4　X 軸位置 5000 からスプリットパルスを開始し、位置 10000 からスプリット長、パルス幅を変えたスプリット出力

論理位置 5000 からスプリットパルスを開始し、論理位置 10000 からスプリット長、パルス幅を変更して残りのパルス数分、スプリットパルスを出力する例です。同期動作の機能を用いて行います。

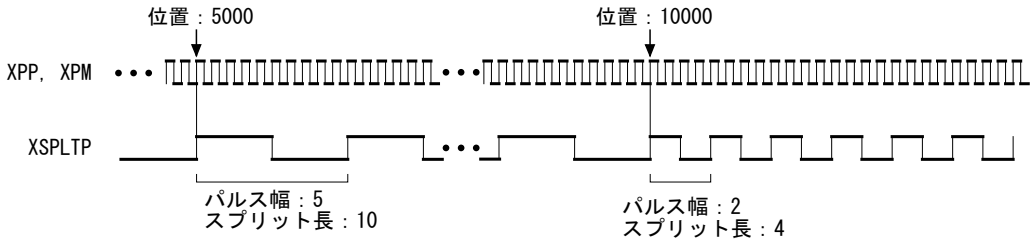


図 2.7-6　ドライブ中に指定位置からスプリット長、パルス幅を変更

【プログラム例】

```
// ドライブの設定(1000 PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                  // 初速度  8M PPS (最大値)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ                  // ドライブ速度  1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                  // 論理位置カウンタ  0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

WR6 ← 2EE0h ﾗｲﾄ                  // 移動パルス数  12000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

// スプリットパルスの設定
// スプリット長、パルス幅の設定
WR6 ← 000Ah ﾗｲﾄ                  // スプリット長  10
WR7 ← 0005h ﾗｲﾄ                  // パルス幅  5
WR0 ← 0117h ﾗｲﾄ

// スプリットパルス数の設定
WR6 ← 0320h ﾗｲﾄ                  // スプリットパルス数  800
WR0 ← 0118h ﾗｲﾄ

// スプリットパルス論理、開始パルスの設定
WR6 ← 0800h ﾗｲﾄ                  // D10        0 SPLL：パルス論理  正
                                  // D11        1 SPLBP：開始パルス  有
WR0 ← 0122h ﾗｲﾄ

// 多目的レジスタの設定
// MR0 の設定
WR6 ← 1387h ﾗｲﾄ                  // MR0  4999
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0110h ﾗｲﾄ

// MR1 の設定
WR6 ← 2710h ﾗｲﾄ                  // MR1  10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0111h ﾗｲﾄ

// MR2 の設定
WR6 ← 0004h ﾗｲﾄ                  // スプリット長  4
WR7 ← 0002h ﾗｲﾄ                  // パルス幅  2
WR0 ← 0112h ﾗｲﾄ

// 多目的レジスタモード設定
WR6 ← 0000h ﾗｲﾄ                  // D1,D0      00 M0T1,0：MR0 の比較対象  論理位置カウンタ
                                  // D3,D2      00 M0C1,0：MR0 の比較条件  ≧
                                  // D5,D4      00 M1T1,0：MR1 の比較対象  論理位置カウンタ
                                  // D7,D6      00 M1C1,0：MR1 の比較条件  ≧
WR0 ← 0120h ﾗｲﾄ
```

```
// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0171h ﾗｲﾄ                  // D3～D0     0001 PREV3～0   : 同期動作の起動要因  MRm 比較が真に変化した
                                 // D8～D4     10111 ACT4～0   : 同期動作の動作  スプリットパルス開始
WR0 ← 0126h ﾗｲﾄ

// 同期動作 SYNC1 設定
WR6 ← 0201h ﾗｲﾄ                  // D3～D0     0001 PREV3～0   : 同期動作の起動要因  MRm 比較が真に変化した
                                 // D8～D4     00000 ACT4～0   : 同期動作の動作  NOP
                                 // D11～D9    001 SNC+3,2,1    : 他 SYNC 起動  SNC+1 起動
WR0 ← 0127h ﾗｲﾄ

// 同期動作 SYNC2 設定
WR6 ← 0030h ﾗｲﾄ                  // D3～D0     0001 PREV3～0   : 同期動作の起動要因  NOP
                                 // D8～D4     00011 ACT4～0   : 同期動作の動作  ロード MRm → SP1
                                 // D11～D9    001 SNC+3,2,1   : 他 SYNC 起動  SNC+1 起動
WR0 ← 0127h ﾗｲﾄ

// SYNC2～0 有効
WR0 ← 0187h ﾗｲﾄ

// ドライブ開始
WR0 ← 0150h ﾗｲﾄ                  // 相対位置ドライブ開始
```

この例では、位置が 4999 になったタイミングでスプリットパルスが開始される設定にすると、実際は 5000 のパルスから開始されます。

【注意】変更のタイミングによっては、変更前後のスプリットパルスが設定値に沿わない動作になる場合がありますので、同期動作などを用いてスプリットパルス動作中にスプリット長やパルス幅を変更する際には注意が必要です。

2.8 汎用入出力信号

本ICは、各軸に8点の汎用入出力信号(nPIO7～0)を持っています。

また、特定の機能を持つ入力信号を、入力信号の持つ機能として使用しない場合、機能を無効設定にすることで汎用入力信号として使用することができます。

2.8.1 nPIOm 信号

nPIOm信号は、以下に示すように様々な目的で入出力信号として使用することができます。

1) 汎用入力信号
2) 汎用出力信号
3) 同期動作の起動要因としての入力信号
4) 同期動作の動作(Action)としての同期パルス出力信号
5) ドライブ状態を出力する出力信号
6) 多目的レジスタとの比較結果を出力する出力信号
7) 外部信号によるドライブ操作のための入力信号

■ nPIOm 信号の機能設定

nPIOm信号の機能設定は、PIO信号設定1命令(21h)で行います。

	D15	D14	D13	D12 (H)	D11	D10	D9	D8	D7	D6	D5	D4 (L)	D3	D2	D1	D0
WR6	P7M1	P7M0	P6M1	P6M0	P5M1	P5M0	P4M1	P4M0	P3M1	P3M0	P2M1	P2M0	P1M1	P1M0	P0M1	P0M0
	nPI07信号		nPI06信号		nPI05信号		nPI04信号		nPI03信号		nPI02信号		nPI01信号		nPI00信号	

WR6レジスタの各nPIOm信号に対応する2ビットを使用目的に応じて設定します。
各nPIOm信号の2ビットの設定値に対応する機能を下表に示します。

表 2.8-1 nPIOm 信号の機能設定

(k:0～7)

PkM1 ビット	PkM0 ビット	機能
0	0	汎用入力 nPI07～0 信号が入力状態になります。 同期動作では、信号の↑や↓で同期動作を起動することができます。 外部ドライブ操作では、nPI04,5 信号によって相対位置ドライブや連続パルスドライブを起動することができます。
0	1	汎用出力 nPI07～0 信号は出力状態になります。
1	0	ドライブ状態出力 nPI07～0 信号は出力状態になり、ドライブ状態を出力します。
1	1	同期パルス・MRm 比較出力 nPI07～0 信号は出力状態になります。nPI03～0 は同期パルスを、nPI07～4 は MRm 比較値を出力します。

■ nPIOm 信号の読み出し

nPIOm信号の信号レベルは、入力／出力にかかわらず、RR4、RR5レジスタで常に読み出すことができます。X軸がRR4レジスタのD7～D0ビット(XPIO7～XPIO0)、Y軸がRR4レジスタのD15～D8ビット(YPIO7～YPIO0)、Z軸がRR5レジスタのD7～D0ビット(ZPIO7～ZPIO0)、U軸がRR5レジスタのD15～D8ビット(UPIO7～UPIO0)となります。それぞれ信号がLowレベルのときは0、Hiレベルのときは1を示します。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
RR4	YPIO7	YPIO6	YPIO5	YPIO4	YPIO3	YPIO2	YPIO1	YPIO0	XPIO7	XPIO6	XPIO5	XPIO4	XPIO3	XPIO2	XPIO1	XPIO0

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
RR5	UPIO7	UPIO6	UPIO5	UPIO4	UPIO3	UPIO2	UPIO1	UPIO0	ZPIO7	ZPIO6	ZPIO5	ZPIO4	ZPIO3	ZPIO2	ZPIO1	ZPIO0

■ 汎用入力

入力信号の機能として、汎用入力信号、同期入力信号、外部信号によるドライブ操作の入力信号の3種類があります。
使用するnPIOm信号の該当2ビットを0,0にし、PIO信号設定1命令(21h)で設定します。

汎用入力信号として使用

nPIO7～0信号の信号レベルがRR4、RR5レジスタに表示されます。X軸がRR4レジスタのD7～D0ビット(XPIO7～XPIO0)、Y軸がRR4レジスタのD15～D8ビット(YPIO7～YPIO0)、Z軸がRR5レジスタのD7～D0ビット(ZPIO7～ZPIO0)、U軸がRR5レジスタのD15～D8ビット(UPIO7～UPIO0)となります。信号がLowレベルのときは0、Hiレベルのときは1を示します。

同期入力信号として使用

nPIOm信号の入力変化を同期動作の起動要因として使用することができます。
同期動作については、2.6節を参照してください。

外部信号によるドライブ操作のための入力信号として使用

相対位置ドライブや連続パルスドライブを、コマンドではなく、nPIOm信号入力によって起動させることができます。
nPIO4信号、nPIO5信号を用いて行い、これらの信号の入力状態および入力変化によってドライブが起動されます。
外部信号によるドライブ操作については、2.12.1項を参照してください。

■ 汎用出力

使用するnPIOm信号の該当2ビットを0,1にし、PIO信号設定1命令(21h)で設定します。

nPIOm信号への書き込みは、WR4、WR5レジスタへの書き込みで行います。X軸がWR4レジスタのD7～D0ビット(XPIO7～XPIO0)、Y軸がWR 4レジスタのD15～D8ビット(YPIO7～YPIO0)、Z軸がWR 5レジスタのD7～D0ビット(ZPIO7～ZPIO0)、U軸がWR 5レジスタのD15～D8ビット(UPIO7～UPIO0)となります。それぞれのビットに書き込んだ値が、各軸のPIO7～0信号に出力されます。各ビットへ0を書き込むとLowレベル、1を書き込むとHiレベルが出力されます。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR4	YPIO7	YPIO6	YPIO5	YPIO4	YPIO3	YPIO2	YPIO1	YPIO0	XPIO7	XPIO6	XPIO5	XPIO4	XPIO3	XPIO2	XPIO1	XPIO0

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR5	UPIO7	UPIO6	UPIO5	UPIO4	UPIO3	UPIO2	UPIO1	UPIO0	ZPIO7	ZPIO6	ZPIO5	ZPIO4	ZPIO3	ZPIO2	ZPIO1	ZPIO0

■ ドライブ状態出力

ドライブ中の状態をnPIOm信号に出力させることができます。
使用するnPIOm信号の該当2ビットを1,0にし、PIO信号設定1命令(21h)で設定します。

ドライブ中、加速中、減速中などのドライブ状態がnPIOm信号から出力されます。
ドライブ状態の出力については、2.12.7項を参照してください。

■ 同期パルス・MRm 比較出力

使用するnPIOm信号の該当2ビットを1,1にし、PIO信号設定1命令(21h)で設定します。

同期パルス出力信号として使用

同期動作の動作(Action)としてnPIO0～nPIO3信号に同期パルスを出力させることができます。
同期動作については、2.6節を参照してください。

MRm 比較出力信号として使用

MRmレジスタとの比較結果をnPIOm信号に出力させることができます。
MR0～MR3比較出力が、nPIO4～nPIO7信号から出力されます。
MRmレジスタについては、2.4節を参照してください。

2.8.2 その他の入力信号

下表に示す、nPIOm信号以外の入力信号についても、その信号の持つ機能を使わない場合には、汎用入力信号として使用することができます。
入力信号の信号レベルは、RR3 レジスタ ページ 0 に表示されます。信号が Low レベルのときは 0、Hi レベルのときは1を示します。

下表に汎用入力信号として用いることができる入力信号を示します。

表 2.8-2 汎用入力信号として用いることができる入力信号

入力信号 (端子番号)	入力信号の持つ機能	状態を表示する各軸の RR3レジスタ ページ0のビット
XSTOP0(74) YSTOP0(93) ZSTOP0(112) USTOP0(131)	ドライブ停止信号	D0ビット(STOP0)
XSTOP1(73) YSTOP1(92) ZSTOP1(111) USTOP1(130)	ドライブ停止信号	D1ビット(STOP1)
XSTOP2(70) YSTOP2(91) ZSTOP2(110) USTOP2(129)	ドライブ停止信号	D2ビット(STOP2)
XECA(45) YECA(47) ZECA(49) UECA(51)	エンコーダＡ相信号	D3ビット(ECA)
XECB(46) YECB(48) ZECB(50) UECB(52)	エンコーダＢ相信号	D4ビット(ECB)
XINPOS(66) YINPOS(85) ZINPOS(104) UINPOS(123)	サーボ用位置決め完了信号	D5ビット(INPOS)
XALARM(67) YALARM(86) ZALARM(105) UALARM(124)	サーボ用アラーム信号	D6ビット(ALARM)
XLMTP(68) YLMTP(87) ZLMTP(106) ULMTP(127)	＋方向ハードリミット信号	D7ビット(LMTP)
XLMTM(69) YLMTM(88) ZLMTM(109) ULMTM(128)	－方向ハードリミット信号	D8ビット(LMTM)

RR3 状態表示
0:Lowレベル
1:Hiレベル

2.9 タイマー

本 IC は、各軸にタイマーを 1 つ内蔵しています。1 μ sec 単位で、1 ～ 2,147,483,647 μ sec の範囲が設定可能です(CLK=16MHz 時)。本 IC の持つ同期動作機能を使用することにより、モータ・ドライブとタイマー機能を組み合わせた様々な動作を精度よく行うことができます。下記にその一例を示します。

■ ドライブ終了後、指定時間後にドライブを開始する。

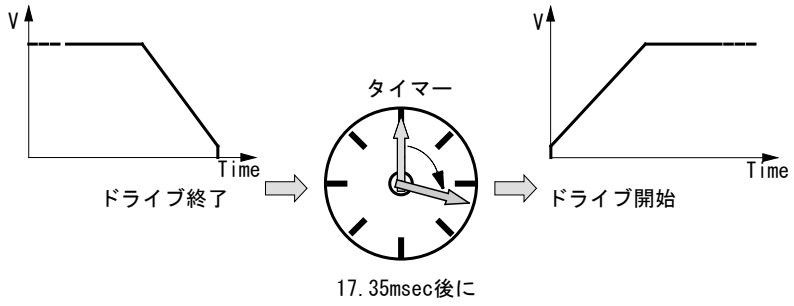


図 2.9-1 タイマー動作 例 1

■ 正確な時間周期ごとに決められたドライブパルスを出力する。

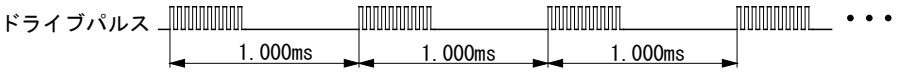


図 2.9-2 タイマー動作 例 2

■ 加減速ドライブにおいて指定の時間だけ定速ドライブを行った後に、減速停止させる。

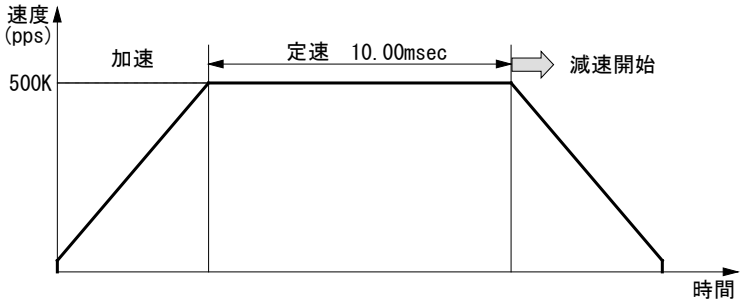


図 2.9-3 タイマー動作 例 3

2.9.1 タイマーの動作

本ICは、31bit長のタイマーカウンタを持っています。タイマーが起動されると、タイマーカウンタは1 μ sec間隔で、0からカウントアップして行き、指定のタイマー値に達する(タイムアップ)と停止します。タイマー動作モードを"1回"に指定した場合にはタイムアップするとタイマー動作は終了します。動作モードを"繰り返し"に指定すると、タイムアップ後にカウンタは再び0からカウントアップして行きます。タイマー停止命令もしくは同期動作からの停止がかかるまで繰り返して動作し続けます。
タイマーのタイムアップは同期動作の起動要因として設定でき、ドライブの開始、外部信号の出力など様々な動作を行わせることができます。同期動作については、2.6節を参照してください。
また、タイムアップ時に割り込み信号を発生させることもできますので、CPU側でタイミングを合わせた処理を行うことも可能です。

2.9.2 タイマーの設定

タイマーを動作させるには、タイマー値と動作モード（1回／繰り返し）を設定する必要があります。

■ タイマー値の設定

タイマー値はタイマー値設定命令（16h）で設定します。WR6,7レジスタに値を設定し、タイマー値設定命令(16h)をWR0レジスタに書き込むと設定されます。タイマー値の単位はμsecで、1～2,147,483,647の範囲で設定します（7.2.23項参照）。

タイマー動作中でも、タイマー値を変更することができます。

■ タイマーのモード設定

タイマーの動作モードは、WR3 レジスタの D14 ビット(TMMD)に設定します。
D14 ビット（TMMD）に 0 を設定すると 1 回、1を設定すると繰り返しの動作になります。

2.9.3 タイマーの始動と停止

■ タイマーの始動

タイマーの始動は、タイマー始動命令（73h）を発行するか、同期動作の動作（Action）にタイマー始動コードを設定し、その同期動作を起動することによって行います。

■ タイマーの停止

タイマーは、動作モードが 1 回の場合には指定のタイマー値に達する（タイムアップ）と停止します。タイマー作動中には、タイマー停止命令（74h）を発行するか、同期動作によって停止させることができます。
動作モードが繰り返しの場合には、タイマー停止命令（74h）を発行するか、同期動作によって停止させます。

2.9.4 タイマーと同期動作

タイマーを同期動作で用いる事ができます。
同期動作の起動要因として、「タイムアップ」を指定できます。同期動作の動作(Action)として、「現在タイマー値を多目的レジスタにセーブ」、「タイマー始動」、「タイマー停止」の 3 種類を指定できます。これらの機能の詳細は、2.6 節を参照してください。

2.9.5 タイマー動作状態と現在タイマー値の読み出し

■ 現在タイマー値の読み出し

現在タイマー値読み出し命令(38h)で動作中の現在タイマー値を読み出すことができます。
タイマーカウンタは、0 からカウントアップして行きます。動作中はいつでもこのタイマーカウンタの値を読み出すことができます。

タイマーカウンタは、タイマー動作が停止すると 0 にクリアされます。タイマー終了後、あるいはタイマー停止命令発行後に現在タイマー値を読み出すと、0 が読み出されます。

■ タイマー動作確認

RR3 レジスタ ページ1の D10 ビット（TIMER）でタイマー動作中を確認することができます。タイマーが起動すると、D10 ビット(TIMER)が1になり、タイマーが動作中であることを示します。

2.9.6 タイマーによる割り込み発生

タイマーがタイムアップしたときに割り込み信号を発生させることができます。WR1レジスタのD9（TIMER）ビットを1に設定します。
割り込み機能については2.10節を参照してください。

2.9.7 タイマーの実例

■ 例 1 X 軸ドライブ終了後、17.35msec 後にドライブを開始する

相対位置ドライブ終了後、17.35msec 後に、再度同じ相対位置ドライブを起動させる例です。同期動作の機能を用いて行います。

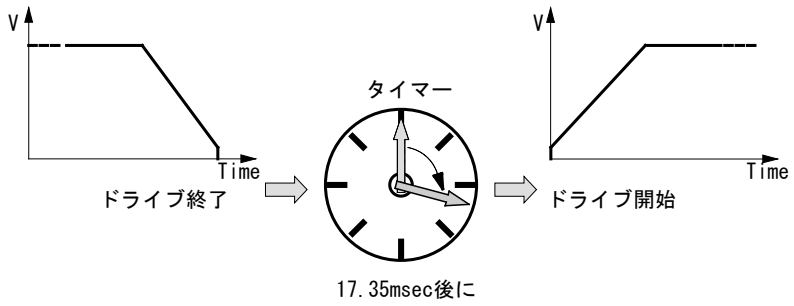


図 2.9-4 タイマー動作 実例 1

【プログラム例】

```
// 加減速ドライブの設定
WR6 ← 0190h ﾗｲﾄ                  // 初速度  400 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 9C40h ﾗｲﾄ                  // ドライブ速度  40K PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← E848h ﾗｲﾄ                  // 加速度  125K PPS/SEC
WR7 ← 0001h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

WR6 ← 9C40h ﾗｲﾄ                  // 移動パルス数  40000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

// タイマーの設定
// 単一タイマー
WR0 ← 011Fh ﾗｲﾄ                  // X軸選択
WR3 ← 0000h ﾗｲﾄ                  // D14        0 TMMD :  タイマー動作  単一

// タイマー値の設定
WR6 ← 43C6h ﾗｲﾄ                  // タイマー値  17350μsec
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0116h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0156h ﾗｲﾄ                  // D3～D0     0110 PREV3～0  :同期動作の起動要因  ドライブ停止
                                  // D8～D4     10101 ACT4～0  :同期動作の動作  タイマー始動
WR0 ← 0126h ﾗｲﾄ

// 同期動作 SYNC1 設定
WR6 ← 00A2h ﾗｲﾄ                  // D3～D0     0010 PREV3～0  :同期動作の起動要因  タイムアップ
                                  // D8～D4     01010 ACT4～0  :同期動作の動作  相対位置ドライブ起動
WR0 ← 0127h ﾗｲﾄ

// SYNC1～0 有効
WR0 ← 0183h ﾗｲﾄ

// ドライブ開始
WR0 ← 0150h ﾗｲﾄ                  // 相対位置ドライブ開始
```

■ 例 2　X 軸に対し 1msec 毎に決められたドライブパルスを出力する

1msec 毎に相対位置ドライブ(20kpps×10 パルスの定速ドライブ)を繰り返し行う例です。同期動作の機能を用いて行います。

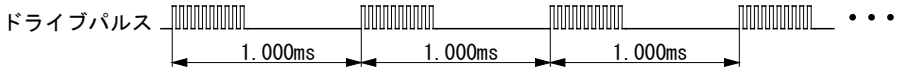


図 2.9-5　タイマー動作　実例２

【プログラム例】

```
// ドライブの設定(1000 PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                  // 初速度　8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 4E20h ﾗｲﾄ                  // ドライブ速度　20K PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 000Ah ﾗｲﾄ                  // 移動パルス数　10
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ                  // 論理位置カウンタ　0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0109h ﾗｲﾄ

// タイマーの設定
// 繰り返しタイマー
WR0 ← 011Fh ﾗｲﾄ                  // X軸選択
WR3 ← 4000h ﾗｲﾄ                  // D14          1 TMMD :   タイマー動作　繰り返し

// タイマー値の設定
WR6 ← 03E8h ﾗｲﾄ                  // タイマー値　1000μsec
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0116h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0153h ﾗｲﾄ                  // D3～D0      0011 PREV3～0   :同期動作の起動要因　ドライブ開始
                                  // D8～D4      10101 ACT4～0   :同期動作の動作　タイマー始動
                                  // D15               0 REP     :同期動作の繰り返し　無効
WR0 ← 0126h ﾗｲﾄ

// 同期動作 SYNC1 設定
WR6 ← 80A2h ﾗｲﾄ                  // D3～D0      0010 PREV3～0   :同期動作の起動要因　タイムアップ
                                  // D8～D4      01010 ACT4～0   :同期動作の動作　ドライブ開始
                                  // D15               1 REP     :同期動作の繰り返し　有効
WR0 ← 0127h ﾗｲﾄ

// SYNC1～0 有効
WR0 ← 0183h ﾗｲﾄ

// ドライブ開始
WR0 ← 0150h ﾗｲﾄ                  // 相対位置ドライブ開始

    ・
    ・
    ・

// タイマー停止
WR0 ← 0174h ﾗｲﾄ                  // タイマー停止

// 同期動作 SYNC1 を無効状態にする
WR0 ← 0192h ﾗｲﾄ                  // 同期動作 SYNC1 を無効状態にする
```

■ 例 3　X 軸加減速ドライブにおいて、10msec 定速ドライブを行った後に、減速停止させる

加減速ドライブ開始後、定速開始から 10msec のタイマーを始動し、タイムアップ後にドライブを減速停止をさせる動作です。同期動作の機能を用いて行います。

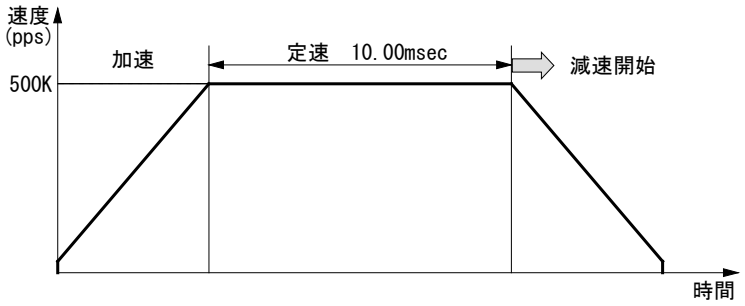


図 2.9-6　タイマー動作　実例３

【プログラム例】

```
// 加減速ドライブの設定
WR6 ← 0064h ﾗｲﾄ                  // 初速度  100 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← A120h ﾗｲﾄ                  // ドライブ速度  500K PPS
WR7 ← 0007h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← E848h ﾗｲﾄ                  // 加速度  125K PPS/SEC
WR7 ← 0001h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

// タイマーの設定
// 単一タイマー
WR0 ← 011Fh ﾗｲﾄ                  // X軸選択
WR3 ← 0000h ﾗｲﾄ                  // D14        0 TMMD :  タイマー動作  単一

// タイマー値の設定
WR6 ← 2710h ﾗｲﾄ                  // タイマー値 10000μsec
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0116h ﾗｲﾄ

// 同期動作の設定
// 同期動作 SYNC0 設定
WR6 ← 0154h ﾗｲﾄ                  // D3～D0     0100 PREV3～0   :同期動作の起動要因  ドライブ定速域開始
                                  // D8～D4     10101 ACT4～0   :同期動作の動作  タイマー始動
WR0 ← 0126h ﾗｲﾄ

// 同期動作 SYNC1 設定
WR6 ← 0112h ﾗｲﾄ                  // D3～D0     0010 PREV3～0   :同期動作の起動要因  タイムアップ
                                  // D8～D4     10001 ACT4～0   :同期動作の動作  ドライブ減速停止
WR0 ← 0127h ﾗｲﾄ

// SYNC1～0 有効
WR0 ← 0183h ﾗｲﾄ

// ドライブ開始
WR0 ← 0152h ﾗｲﾄ                  // +方向連続パルスドライブ開始
```

2.10 割り込み

割り込みの発生は、X，Y，Z，U各軸から発生させる割り込みと、連続補間ドライブ時に発生させる割り込みの2種類があります。
CPUに対する割り込み信号は、X，Y，Z，U各軸から発生させる割り込みのINT0N信号、連続補間時に発生させる割り込みのINT1N信号の2本となります。
割り込み要因はすべて割り込み許可／禁止を設定することができます。リセット時にはすべて禁止状態になります。

2.10.1 X, Y, Z, U軸の割り込み

X, Y, Z, U軸から発生させる割り込み要因は次の通りです。

表 2.10-1 X,Y,Z,U軸から発生させる割り込み要因

許可/禁止の設定 WR1レジスタ	発生の確認 RR1レジスタ	割り込み発生要因
D0 (CMR0)	D0 (CMR0)	多目的レジスタMR0と比較対象との比較結果が、比較条件を満たすように変化した
D1 (CMR1)	D1 (CMR1)	多目的レジスタMR1と比較対象との比較結果が、比較条件を満たすように変化した
D2 (CMR2)	D2 (CMR2)	多目的レジスタMR2と比較対象との比較結果が、比較条件を満たすように変化した
D3 (CMR3)	D3 (CMR3)	多目的レジスタMR3と比較対象との比較結果が、比較条件を満たすように変化した
D4(D-STA)	D4(D-STA)	ドライブが開始した
D5(C-STA)	D5(C-STA)	加減速ドライブで、定速域でのパルス出力を開始した
D6(C-END)	D6(C-END)	加減速ドライブで、定速域でのパルス出力を終了した
D7(D-END)	D7(D-END)	ドライブが終了した
D8(H-END)	D8(H-END)	自動原点出しが終了した
D9(TIMER)	D9(TIMER)	タイマーがタイムアップした
D10(SPLTP)	D10(SPLTP)	スプリットパルスを出力した(正論理の場合、スプリットパルスの↑で発生)
D11(SPLTE)	D11(SPLTE)	スプリットパルスが終了した
D12(SYNC0)	D12(SYNC0)	同期動作SYNC0が起動した
D13(SYNC1)	D13(SYNC1)	同期動作SYNC1が起動した
D14(SYNC2)	D14(SYNC2)	同期動作SYNC2が起動した
D15(SYNC3)	D15(SYNC3)	同期動作SYNC3が起動した

■ 割り込みの設定と読み出し

それぞれの割り込み要因は、上表に示すようにWR1レジスタで許可(1)／禁止(0)を設定します。許可を設定した割り込み要因が真になるとRR1レジスタのその要因に対応するビットが1になり、割り込み出力信号(INT0N)がLowレベルになります。上位CPUがRR1レジスタを読み出すと、RR1レジスタの1の立っているビットは0にクリアされ、割り込み出力信号(INT0N)はHi-Zに戻ります。すなわち、RR1レジスタ読み出し動作により自動的に割り込み信号は解除されます。また、割り込みが発生したことを知らせる情報は、割り込み発生後はじめのRR1レジスタ読み出しにより1回だけCPUに伝えられ、その後RR1レジスタを読み出しでも、次に割り込み要因が真にならない限り、該当するビットは0を示しています(リードリセット方式)。

■ 複数の割り込み

複数の割り込み要因を許可している場合、はじめの割り込み要因が真になると割り込み信号(INT0N)はLowになり、RR1レジスタの対応するビットが1になります。その後、CPUがRR1レジスタを読み出す前に別の要因が真になった場合にも、別の要因に対応するビットが1になります。RR1レジスタを読み出すと複数のビットが1になっており、それぞれの割り込み要因が発生していることを通知します。

■ 8ビットデータバス時の割り込み

8ビットデータバスの時は、WR1Hレジスタ、WR1Lレジスタごとに許可(1)／禁止(0)を設定します。割り込みが発生（割り込み信号(INT0N)がLow）した場合も、RR1HレジスタとRR1Lレジスタをそれぞれ読み出します。もちろん片方のレジスタだけを許可設定している場合には、他方のレジスタは読み出す必要はありません。RR1Hレジスタの1回の読み出しでRR1H内の割り込み発生を示すビットは0にクリアされます。RR1Lレジスタについても同様です。両レジスタの全てのビットがクリアされると割り込み信号(INT0N)はHi-Zに戻ります。

■ I²Cシリアルインターフェイスバス時の割り込み

I²Cシリアルインターフェイスバスの時は、WR1Hレジスタ、WR1Lレジスタごとに許可(1)／禁止(0)を設定できます。WR1レジスタとしてまとめて許可(1)／禁止(0)を設定することも可能です。割り込みが発生（割り込み信号(INT0N)がLow）した場合、RR1レジスタを読み出します。片方のレジスタだけを許可設定している場合でも、片方のレジスタだけを読み出すということはせず、必ず2バイト(RR1L,RR1H)を読み出してください。RR1レジスタを読み出すと、RR1レジスタの1の立っているビットは0にクリアされ、割り込み出力信号(INT0N)はHi-Zに戻ります。I²Cシリアルインターフェイスバスについては、4章を参照してください。

WR1レジスタについては6.5節を、RR1レジスタについては6.12節を参照してください。

CPU リードタイミングの注意

CPUからのリード／ライトサイクルのタイミングは、10.2.2項に示してある通りです。リードサイクルにおいてRDN信号がLowレベルの区間においてはアドレス信号A[3:0]は確定していなければなりません。tAR最小＝0およびtRA最小＝3nsecとなっています。この条件に違反して、RDN信号がLowレベル区間に非有効アドレスデータが入り込んでしまうと、他のレジスタの読み出し動作時にRR1のデータがクリアされ、割り込み信号(INT0N)が解除されてしまう可能性があります。割り込み信号(INT0N)を使用する場合にはCPUリードタイミングに十分ご注意ください。

2.10.2 連続補間の割り込み

補間モード設定命令(2Ah)で許可(1)／禁止(0)を設定します。許可を設定した割り込み要因が真になると、連続補間割り込み出力信号(INT1N)がLowレベルになります。

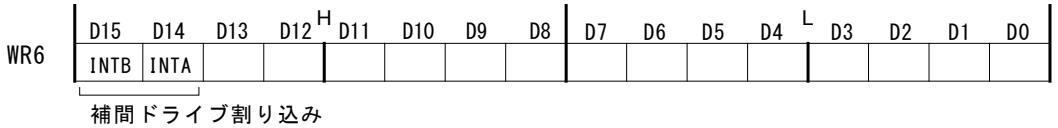


表 2.10-2 連続補間時に発生させる割り込み要因

許可/禁止の設定 補間モード	割り込み発生要因
D14 (INTA)	プリバッファのスタックカウンタが４→３に変化した
D15 (INTB)	プリバッファのスタックカウンタが８→７に変化した

補間割り込み信号(INT1N)信号は、次の条件でクリアされ、Hi-Zに戻ります。

① 補間割り込みクリア命令(6Fh)発行
② 次セグメントの補間実行命令発行
③ 連続補間ドライブ終了

2つの割り込み要因の両方を許可している場合、はじめの割り込み要因が真になると割り込み信号(INT1N)はLowになります。その後、クリアする前にもう一つの要因が真になった場合、割り込み信号(INT1N)はLowのままです。この状態でクリアをすると、補間割り込み信号(INT1N)信号はHi-Zに戻ります。

2.11 入力信号フィルタ

本ICは、IC内部において、各入力信号の入力段に積分型のフィルタを装備しています。図2.11-1はX軸の各入力信号のフィルタ構成を示していますが、Y, Z, U軸についても同様の回路を持っています。フィルタの時定数は、図中のT発振回路によって決まります。本ICでは、2つのフィルタの時定数AとBが存在し、入力信号の種類によってフィルタ時定数AとBのどちらを使用するかが決まっています。フィルタの有効／無効およびフィルタの時定数は、入力信号フィルタモード設定命令(25h)で設定をします。

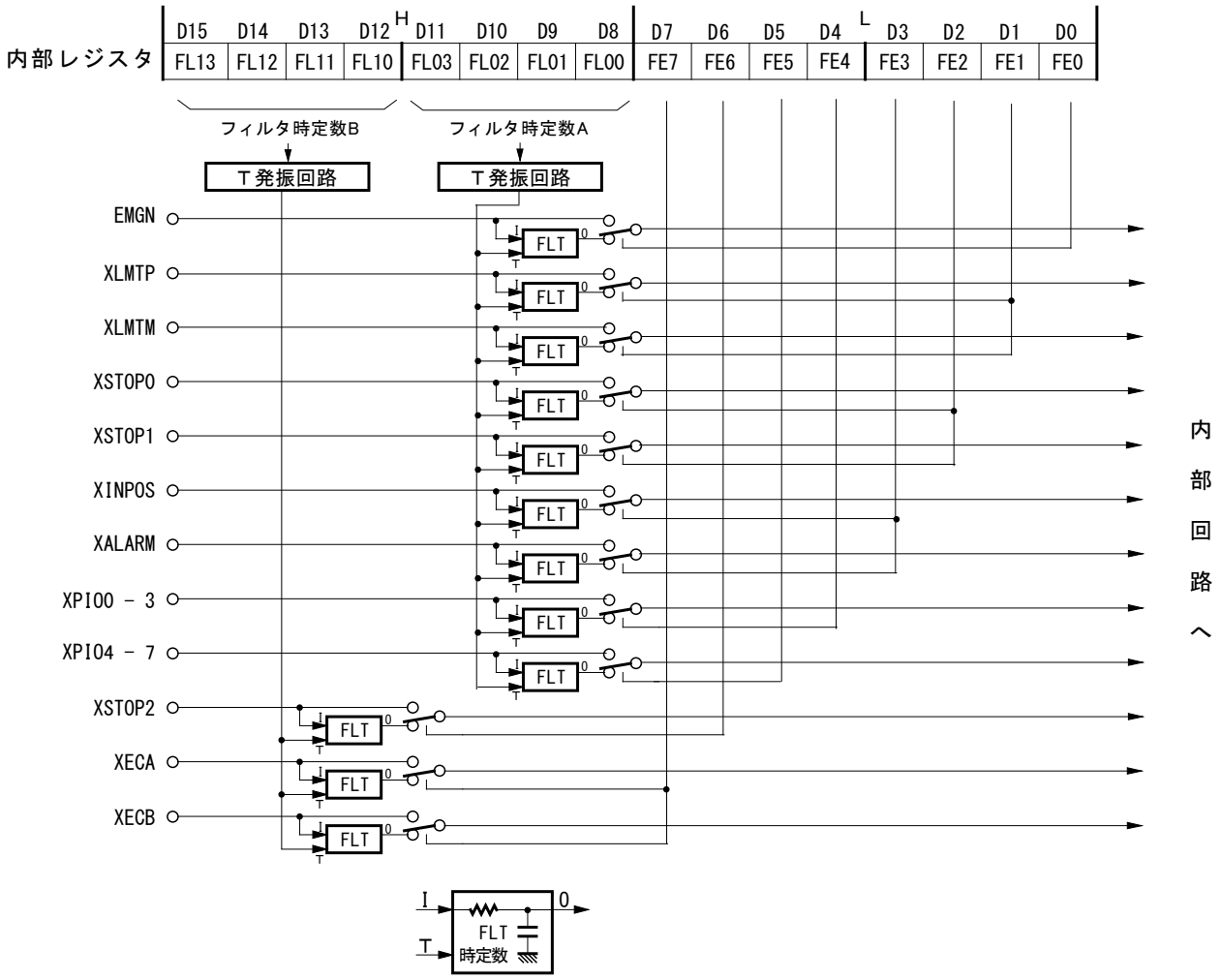


図 2.11-1 X軸入力信号フィルタ回路 概念図

2.11.1 入力信号フィルタ機能の設定

各入力信号のフィルタ機能の設定は、入力信号フィルタモード設定命令(25h)で行います。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6	FL13	FL12	FL11	FL10	FL03	FL02	FL01	FL00	FE7	FE6	FE5	FE4	FE3	FE2	FE1	FE0
	フィルタ時定数B				フィルタ時定数A				各入力信号のフィルタ有効／無効							

各入力信号に対し、IC 内蔵のフィルタ機能を有効にするか、無効(スルー)にするかを D7～0 ビット(FE7～FE0)に設定します。
フィルタ機能を有効にする場合は1を、無効(スルー)にする場合は 0 を設定します。
それぞれのビットに対応する入力信号は、表 2.11-1 のとおりです。各入力信号に適用されるフィルタの時定数 A, B は決まっています。

表 2.11-1 入力信号と対応する時定数

指定ビット	入力信号	適用する時定数
D0(FE0)	EMGN *	フィルタ時定数A
D1(FE1)	nLMTP, nLMTM	
D2(FE2)	nSTOP0, nSTOP1	
D3(FE3)	nINPOS, nALARM	
D4(FE4)	nPI03～0	
D5(FE5)	nPI07～4	
D6(FE6)	nSTOP2	フィルタ時定数B
D7(FE7)	nECA, nECB	

*注：EMGN 信号は X 軸の WR6 レジスタ D0 ビットで設定します。

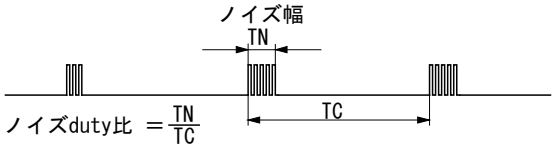
フィルタの時定数 A の設定は D11～8 ビット(FL03～FL00)、フィルタ時定数 B の設定は D15～D12 ビット(FL13～FL10)で行います。
設定できる時定数の値は、表 2.11-2 に示す 16 段階の中から選択します。時定数を上げると除去可能な最大ノイズ幅は上がりますが、信号の遅延時間が大きくなりますので、適切な値を設定します。通常は、時定数 A(FL03～00)は Ah または Bh の値に設定することをお勧めします。時定数 B(FL13～10)はエンコーダ入力信号のために用意されている時定数です。

表 2.11-2 時定数と除去可能な最大ノイズ幅

(CLK=16MHz 時)

時定数 (Hex)	除去可能な最大ノイズ幅*1	入力信号遅延時間
0	437.5ｎsec	500ｎsec
1	875ｎsec	1μsec
2	1.75μsec	2μsec
3	3.5μsec	4μsec
4	7μsec	8μsec
5	14μsec	16μsec
6	28μsec	32μsec
7	56μsec	64μsec
8	112μsec	128μsec
9	224μsec	256μsec
A	448μsec	512μsec
B	896μsec	1.024msec
C	1.792msec	2.048msec
D	3.584msec	4.096msec
E	7.168msec	8.192msec
F	14.336msec	16.384msec

＊1：ノイズ幅



ノイズduty比(信号上にノイズが発生する時間比率)が、いかなる時においても1/4以下であることが条件です。

リセット時には、すべての入力信号フィルタ機能は無効(スルー)になっています。
入力信号EXPLSN、PIN7～0に関しては、フィルタ機能を使用することはできません。

2.11.2 入力信号フィルタの設定例

フィルタの時定数 A に属する入力信号に対し、EMGN と XLMTP, XLMTM, XSTOP0, XSTOP1 入力信号に 128μsec 遅延のフィルタを設定し、他の入力信号はスルーとします。
フィルタの時定数 B に属する XECA, XECB, XSTOP2 入力信号はスルーとします。

【プログラム例】

```
// 入出力信号フィルタモード設定
WR6 ← 0807h ﾗｲﾄ        // D15～D12  0000  フィルタ時定数B  フィルタ遅延:500nsec
                       // D11～D8   1000  フィルタ時定数A  フィルタ遅延:128μsec
                       // D7         0  XECA, XECB信号(フィルタ時定数B):フィルタ無効（スルー）
                       // D6         0  XSTOP2信号(フィルタ時定数B):フィルタ無効（スルー）
                       // D5         0  XPIO4－7信号(フィルタ時定数A):フィルタ無効（スルー）
                       // D4         0  XPIO0－3信号(フィルタ時定数A):フィルタ無効（スルー）
                       // D3         0  XINPOS, XALARM信号(フィルタ時定数A):フィルタ無効（スルー）
                       // D2         1  XSTOP0,1信号(フィルタ時定数A):フィルタ有効
                       // D1         1  XLMTP, XLMTM信号(フィルタ時定数A):フィルタ有効
                       // D0         1  EMGN信号(フィルタ時定数A):フィルタ有効
WR0 ← 0125h ﾗｲﾄ
```

2.12 その他の機能

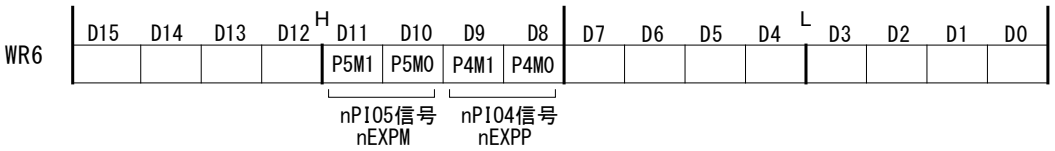
2.12.1 外部信号によるドライブ操作

相対位置ドライブや連続パルスドライブを、コマンドではなく、信号入力（nEXPP, nEXPM）によって、起動する機能です。システムで制御するモータの軸が多くなり、各軸のジョグ送りなどのマニュアル操作を１つのＣＰＵがすべて行おうとすると、ＣＰＵの負担が大きくなり、十分な応答ができなくなる可能性があります。本ＩＣでは、外部信号によるドライブ操作機能によって、これらのＣＰＵの負担を軽減することができます。また手動パルサーのエンコーダ2相信号を入力して、ジョグ送りを行うことができます。
nEXPP, nEXPM信号は、汎用入出力信号nPIO4,5信号が割り当てられています。

外部信号によるドライブ操作を行うには、次の設定を行う必要があります。

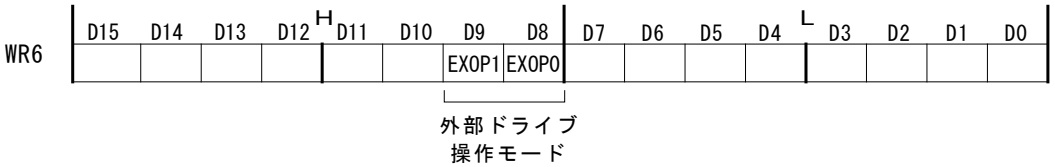
① PIO信号設定1命令(21h)を用いて、nPIO4,5 信号を入力に設定
② PIO信号設定2・その他設定命令(22h)を用いて、ドライブ操作のモード設定

■ nPIOm 信号の外部ドライブ操作機能設定
外部信号によるドライブ操作を行うには、汎用入力信号のnPIO4,5信号を外部ドライブ操作の入力信号（nEXPP, nEXPM）に設定します。
PIO信号設定1命令(21h) のD11～8ビットで設定します。



nPIO4信号の機能を外部ドライブ操作の入力信号（nEXPP）として使用するために、D9,8ビットに0,0を設定します。同様に、nPIO5信号のD11,10ビットに0,0を設定します。

■ ドライブ操作のモード設定
外部ドライブのモードを設定します。
PIO信号設定2・その他設定命令(22h)のD9,8ビットで設定します。



外部入力信号（nEXPP, nEXPM）によるドライブ操作モードはD9,8ビットの2ビットで設定を行います。
各ビットの値に対応するドライブ操作モードは下表のようになります。

表 2.12-1 外部信号によるドライブ操作モード

D9(EXOP1)	D8(EXOP0)	外部信号によるドライブ操作モード
0	0	外部信号によるドライブ操作無効
0	1	連続パルスドライブモード
1	0	相対位置ドライブモード
1	1	手動パルサーモード

■ 相対位置ドライブモード

PIO信号設定2・その他設定命令(22h)のD9,8ビットを1,0にセットし、ドライブに必要な速度パラメータ、移動パルス数(正の値)を設定します。nEXPP信号をHiレベルからLowレベルに落とすと、その↓で＋方向の相対位置ドライブが起動します。nEXPM信号の場合も同様で、HiレベルからLowレベルに落とすと、その↓で－方向の相対位置ドライブが起動します。各入力操作信号のLowレベル幅は、最小4CLKサイクル以上必要です。ドライブが完了しない前に、再度信号を立ち下げても無効になります。

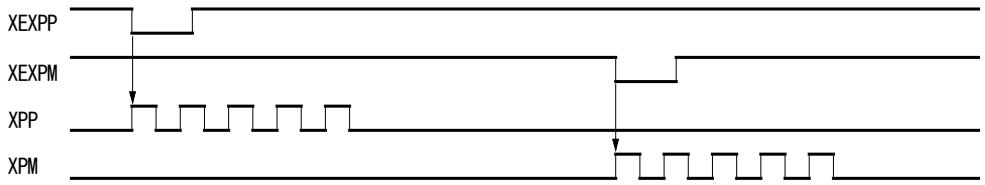


図 2.12-1 外部操作信号による移動パルス数５のＸ軸相対位置ドライブの例

■ 連続パルスドライブモード

PIO信号設定2・その他設定命令(22h)のD9,8ビットを0,1にセットし、ドライブに必要な速度パラメータを設定します。nEXPP信号をHiレベルからLowレベルに落とすと、Lowレベルの期間、連続して＋方向のドライブパルスを出力します。nEXPP信号をLowからHiレベルに戻すと、加減速ドライブのときは減速停止、定速ドライブのときは即停止します。nEXPM信号の場合も、同様にして、－方向のドライブパルスを連続して出力します。ドライブ途中で他方の入力信号をHiレベルからLowレベルに落とすと、現在の方向のドライブが終了すると直ちに他方のドライブを開始します。

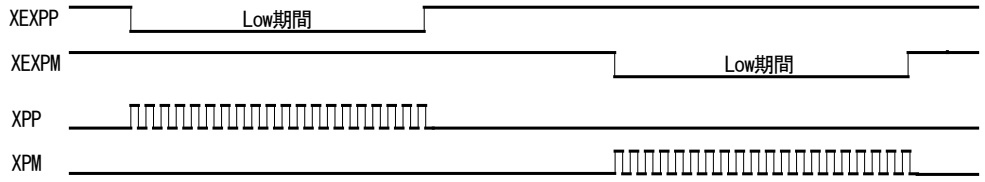


図 2.12-2 外部操作信号によるＸ軸連続パルスドライブの例

■ 手動パルサーモード

PIO信号設定2・その他設定命令(22h)のD9,8ビットを1,1にセットし、ドライブに必要な速度パラメータ、移動パルス数を設定します。エンコーダのA相信号をEXPP入力に、B相信号をnEXPM入力に接続します。nEXPM信号がLowレベルでnEXPP信号の↑で＋方向の相対位置ドライブが起動します。また、nEXPM信号がHiレベルでnEXPP信号の↑で－方向の相対位置ドライブが起動します。移動パルス数の設定が1であればnEXPP信号の↑で1つのドライブパルスを出力します。移動パルス数の設定がTPであればTP個のドライブパルスを出力します。

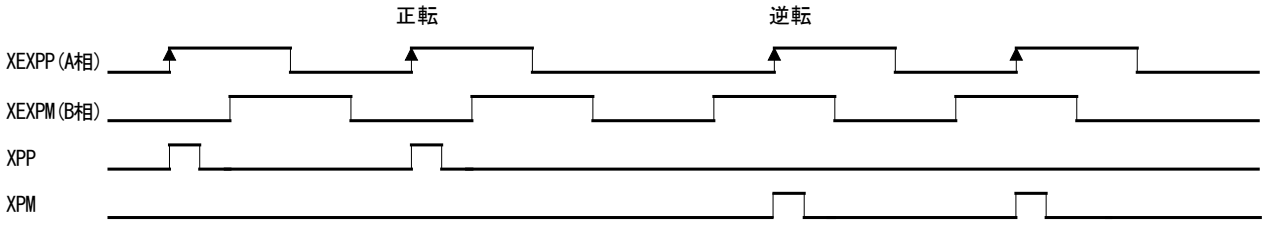


図 2.12-3 手動パルサーによる移動パルス数１のＸ軸ドライブ例

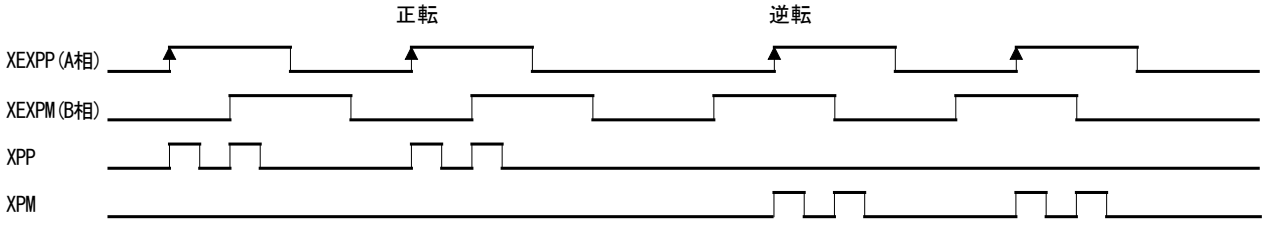


図 2.12-4 手動パルサーによる移動パルス数２のＸ軸ドライブ例

nEXPP信号の↑から次の↑の間にTP個のドライブパルスを出力し終えるために、速度パラメータはつぎの条件で設定します。

$$DV \geqq F \times TP \times 2$$

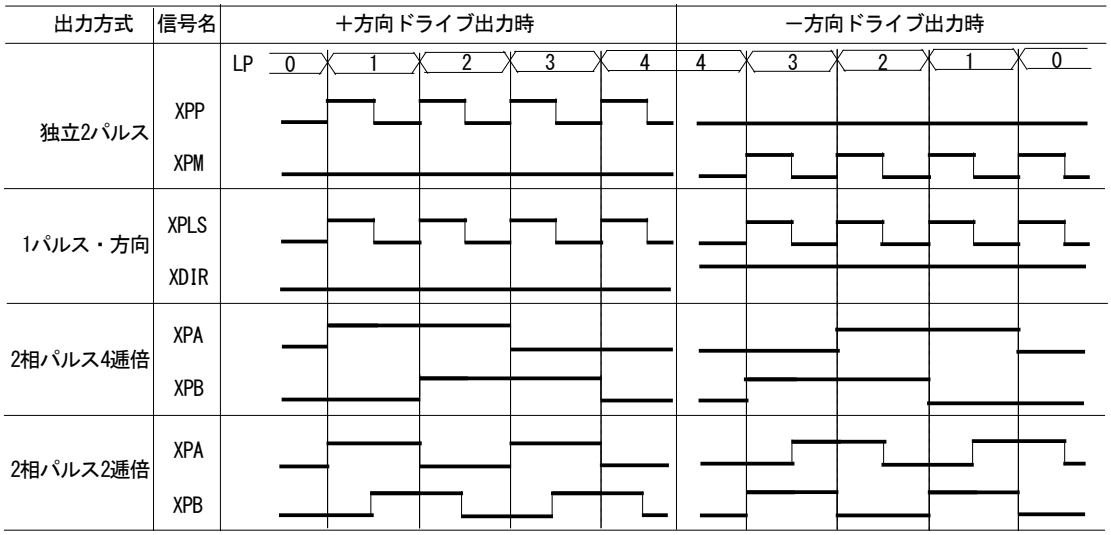
DV:ドライブ速度(pps)
TP:移動パルス数
F:手動パルサーエンコーダの最高速時の周波数(Hz)

例えば、手動パルサーの最高速周波数をF=500Hzとし、移動パルス数をTP=1とすると、ドライブ速度は、DV=1000pps以上の値に設定する必要があります。また、加減速ドライブは行いませんので初速度SVはドライブ速度DV以上の値に設定します。ただし、駆動モータがステッピングモータの場合は、モータの自起動周波数を超えない範囲内でドライブ速度を設定します。

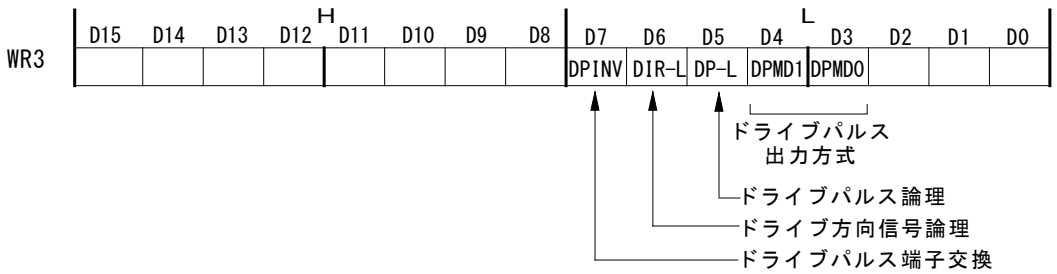
2.12.2 ドライブパルス出力方式の選択

ドライブパルス出力信号は、X軸はXPP/PLS/PA(37)とXPM/DIR/PB(38)、Y軸はYPP/PLS/PA (39)とYPM/DIR/PB(40)、Z軸はZPP/PLS/PA (41)とZPM/DIR/PB(42)、そしてU軸はUPP/PLS/PA (43)とUPM/DIR/PB(44)です。各軸とも下表に示す4つのパルス出力方式を選択することができます。独立2パルス方式では、＋方向ドライブ時にはnPPに、－方向ドライブ時にはnPMにドライブパルスを出力します。また、1パルス・方向方式では、nPLSにドライブパルスを出力し、nDIRには方向信号が出力されます。2相パルス方式にすると出力信号nPAに2相パルスのA相信号が、出力信号nPBに2相パルスのB相信号が出力されます。2相4逓倍では、nPA、nPBパルスの出力が変化したときに論理位置カウンタがアップ(ダウン)します。2相2逓倍では、nPAパルスの出力が変化したときに論理位置カウンタがアップ(ダウン)します。

表 2.12-2 X軸ドライブパルス出力方式の例



ドライブパルス出力方式は、WR3レジスタのD4,3ビット(DPMD1,0)で設定します。



各ビットの値に対応するドライブ操作モードは下表のようになります。

表 2.12-3 ドライブパルス出力方式

D4(DPMD1)	D3(DPMD0)	ドライブパルス出力方式
0	0	独立２パルス方式
0	1	１パルス・方向方式
1	0	２相パルス４逓倍方式
1	1	２相パルス２逓倍方式

1パルス・方向方式の場合は、パルス信号nPLSと方向信号nDIRが出力されるタイミングを、11.2節で確認してください。ドライブ前にnDIR信号を設定したい場合は、方向信号＋設定命令(58h)または方向信号－設定命令(59h)を発行します。

また、D5ビット(DP-L)でドライブパルスのパルス論理、D6ビット(DIR-L)でドライブ方向信号(DIR)論理、D7ビット(DPINV)でドライブパルス信号の端子交換を設定することができます。

【注意】補間ドライブではドライブ途中で方向が変わるため、1 パルス・方向方式は適しません。補間ドライブを行う場合は、独立 2 パルス方式を使用してください。

2.12.3 エンコーダ入力パルス方式の選択

実位置カウンタのアップ／ダウンカウント入力信号となるエンコーダパルス入力(nECA/PPIN、nECB/PMIN)は、2相パルス入力とアップ／ダウンパルス入力から選択することができます。

■ 2 相パルス入力

2 相パルス入力方式として、2 相 4 逓倍、2 逓倍、1 逓倍の 3 種類から選択できます。正論理パルスでA相が進んでいるときはカウントアップし、B相が進んでいるときはカウントダウンします。4 逓倍設定時は両信号の↑、↓でカウントアップ、ダウンします。2 逓倍設定時は A 相信号の↑、↓でカウントアップ、ダウンします。1 逓倍設定時は、B 相が Low の状態で A 相信号の↑でカウントアップ、A 相信号の↓でカウントダウンします。

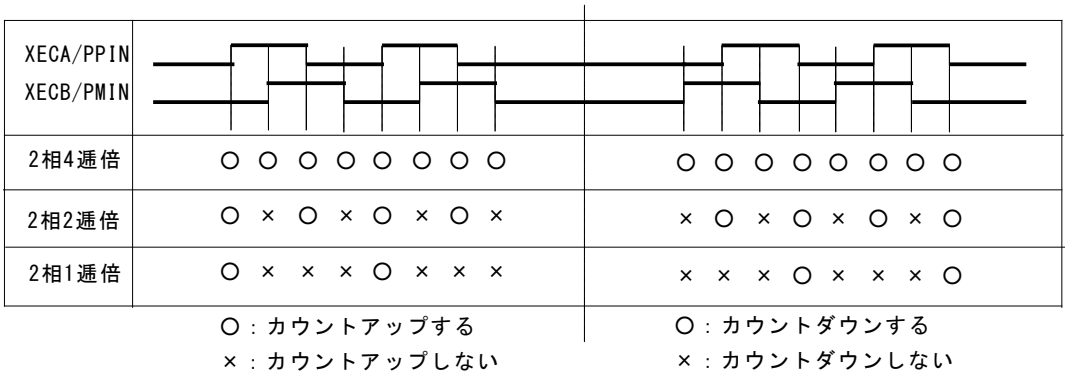


図 2.12-5 X軸 2 相パルス入力の例

■ アップ／ダウンパルス入力

nECA/PPIN がカウントアップ入力に、nECB/PMIN がカウントダウン入力になります。それぞれ、正パルスの↑でカウントします。(正論理設定時)

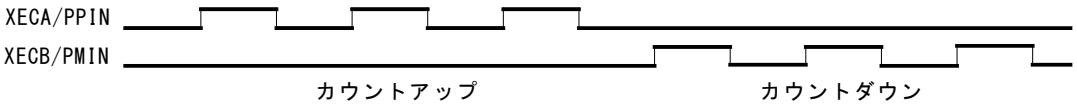
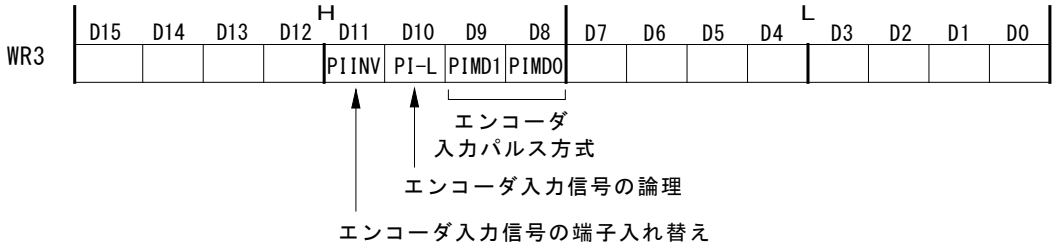


図 2.12-6 X軸アップ／ダウンパルス入力の例

■ エンコーダ入力パルス方式の設定

エンコーダ入力パルス方式は、WR3 レジスタの D8,9 ビット(PIMD0,1)で設定します。



各ビットの値に対応するエンコーダ入力パルス方式は下表のようになります。

表 2.12-4 エンコーダ入力パルス方式

D9(PIMD1)	D8(PIMD0)	エンコーダ入力パルス方式
0	0	２相パルス入力 ４逓倍
0	1	２相パルス入力 ２逓倍
1	0	２相パルス入力 １逓倍
1	1	アップ／ダウンパルス入力

また、D10 ビット(PI-L)でエンコーダ入力信号の論理、D11 ビット(PIINV)でエンコーダ入力信号の端子入れ替えを設定できます。
エンコーダ入力信号の端子入れ替えによる実位置カウンタの増減は次のようになります。

表 2.12-5 エンコーダ入力信号の端子入れ替えの有無による実位置カウンタの増減

WR3/D11(PIINV)	入力パルスモード	実位置カウンタの増減
0	２相モード	Ａ相が進んでいるときカウントアップする。 Ｂ相が進んでいるときカウントダウンする。
	アップダウンパルスモード	nPPIN パルス入力のときカウントアップする。 nPMIN パルス入力のときカウントダウンする。
1	２相モード	Ｂ相が進んでいるときカウントアップする。 Ａ相が進んでいるときカウントダウンする。
	アップダウンパルスモード	nPMINパルス入力のときカウントアップする。 nPPIN パルス入力のときカウントダウンする。

2.12.4 ハードリミット信号

ハードウェアリミット信号(nLMTP, nLMTM)は、＋方向、－方向のドライブパルスをそれぞれ抑止する信号入力です。

リミット信号の有効／無効、リミット信号の論理レベル、リミット信号がアクティブになったとき減速停止させるか即停止させるか、およびリミット信号の端子交換を選択することができます。

リミット信号の有効／無効、論理レベル、停止方式はWR2レジスタのD12～10ビットで設定します。WR2レジスタの設定については、6.6節を参照してください。
リミット信号の端子交換の選択は、WR3 レジスタの D12 ビット(LMINV)で設定します。WR3 レジスタの設定については、6.7 節を参照してください。

リミット信号は、RR3レジスタ ページ0でその状態を常に読み出すことができます。

2.12.5 サーボモータドライバ対応の信号

■ nINPOS 信号、nALARM 信号

サーボモータドライバとの接続のための入力信号として、インポジション（位置決め完了）信号を入力するnINPOSと、アラーム信号を入力するnALARMがあります。

各々の信号は有効／無効および論理レベルを設定することができます。設定はWR2レジスタのD9～6ビットで設定します。WR2レジスタの設定については、6.6節を参照してください。

nINPOS入力信号は、サーボモータドライバのインポジション（位置決め完了）信号に対応します。有効に設定すると、ドライブ終了後、nINPOS入力信号がアクティブになるのを待ってから、RR0主ステータスレジスタのD3～0ビット(n-DRV:ドライブ状態)が0に戻ります。

nALARM入力信号は、サーボモータドライバからのアラーム信号を受信します。有効に設定すると、ドライブ中、nALARM入力信号を常に監視し、アクティブ状態になるとドライブを即停止します。このとき、RR2レジスタのD4(ALARM)ビットおよびD14(ALARM)ビットに1が立ちます。

これらのサーボモータドライバ用入力信号は、RR3レジスタ ページ0でその状態を常に読み出すことができます。

■ 偏差カウンタクリア出力信号

サーボモータドライバ用出力信号として、偏差カウンタクリア出力信号（nDCC）があります。
偏差カウンタクリア出力信号（nDCC）の論理、パルス幅を設定することができます。設定は自動原点出しモード設定2命令(24h)のD3～6ビットで設定します。自動原点出しモード設定2命令(24h)については、7.3.5項を参照してください。

偏差カウンタクリア出力命令(72h)を発行すると、自動原点出しモード設定2命令(24h)で設定した、パルスの論理レベル、パルス幅をもとに、nDCC出力端子から偏差カウンタクリアパルスを出力します。

自動原点出しで偏差カウンタクリア出力信号（nDCC）を使用する場合は、2.5.2項、2.5.4項を参照してください。

2.12.6 緊急停止

本ICは、4軸すべてのドライブを緊急停止させるための入力信号として、EMGN信号があります。EMGN信号は、通常Hiレベルにしておきます。Lowレベルに落とすと、ドライブ中の全軸が即停止し、各軸のRR2レジスタのD5(EMG)ビットおよびD15(EMG)ビットが1になります。EMGN信号は、論理レベルを選択することができませんので、ご注意ください。

CPU側から4軸に対して緊急停止をかけるには、次の方法があります。

a. 4軸に対して同時に**即停止命令を発行**
   WR0レジスタに、4軸すべてを指定して、即停止命令(57h)を書き込みます。
b. **コマンドリセットを発行**
   WR0レジスタに、00FFhを書き込むとリセットがかかります。

2.12.7 ドライブ状態の出力

ドライブ中／停止の状態は、RR0レジスタのD3～D0(n-DRV)ビットと、nPIO0信号に出力されます。
ドライブ中のドライブ速度の加速／定速／減速の状態は、各軸のRR3レジスタ ページ1のD4(ASND), D5(CNST), D6(DSND)ビットと、nPIO2/ASND, nPIO3/CNST, nPIO4/DSND信号に出力されます。

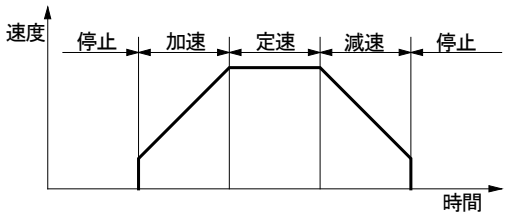


図 2.12-7 ドライブの状態

表 2.12-6 ドライブ状態に対応した RR0、RR3 レジスタおよび nPIOm 信号

ドライブ状態	RR0 レジスタ	RR3 レジスタ ページ 1			nPIOm 信号			
	D3～0(n-DRV)	D4/ASND	D5/CNST	D6/DSND	nPIO0/DRIVE	nPIO2/ASND	nPIO3/CNST	nPIO4/DSND
停止	0	0	0	0	Low	Low	Low	Low
加速	1	1	0	0	Hi	Hi	Low	Low
定速	1	0	1	0	Hi	Low	Hi	Low
減速	1	0	0	1	Hi	Low	Low	Hi

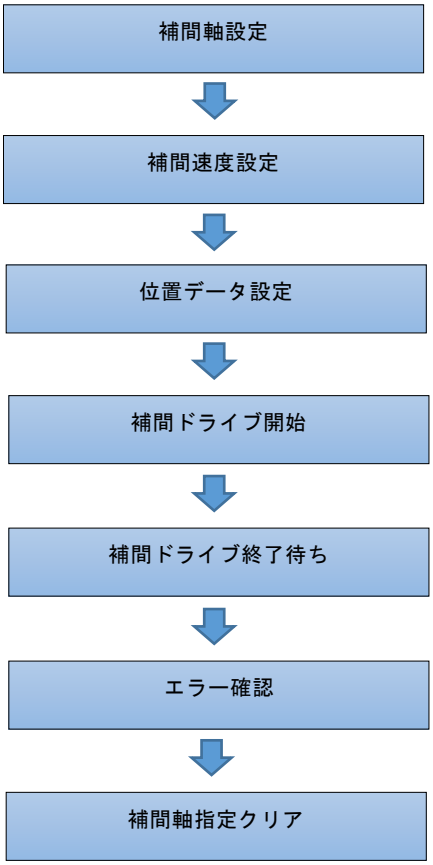
また、S字加減速ドライブにおける加速度、減速度の増加／一定／減少の状態も、各軸のRR3レジスタ ページ1のD7(AASND), D8(ACNST), D9(ADSND)ビットと、nPIO5/AASND, nPIO6/ACNST, nPIO7/ADSND信号に出力されます。

nPIOm信号にドライブ状態を出力させるには、PIO信号設定1命令(21h)で行います。7.3.2項を参照してください。

3. 補間

補間ドライブとは、2 軸以上の軸がそれぞれ1ドライブパルス毎に位置の補間を取りながら移動する動作のことです。
本ICは、4軸中の任意の軸を選択し、直線補間、円弧補間、ヘリカル補間、ビットパターン補間ドライブを行うことができます。また、本ICを複数チップ使用して5軸以上の多軸直線補間を行うこともできます。

補間動作を行うための基本的な操作手順は次のようになります。



補間軸設定
補間を行う軸を指定します。補間モード設定命令(2Ah)で行います。
補間モード設定命令(2Ah)については、7.3.8項を参照してください。

【注意】補間軸の指定(補間モード設定命令(2Ah)の発行)は、必ず補間動作の設定の最初に行います。補間速度や位置データの設定より後に行うと、補間ドライブは正しく実行されません。

補間速度設定
補間ドライブの速度を設定します。速度の設定は、指定された軸の中からX>Y>Z>Uの優先順位で自動的に決まる主軸に対して行います。例えば、補間を行う軸としてX,Z,Uの3軸を指定した場合には、主軸がX軸となります。初速度、ドライブ速度などの速度パラメータを主軸に設定します。主軸は、補間ドライブが開始されると補間演算部に対して主軸パルスを出力します。補間演算部ではこの主軸パルスのタイミングで演算サイクルが行われ、補間を行う各軸のドライブパルスが生成されます。図1.2-1 MCX514全体機能ブロック図を参照してください。主軸パルスは補間演算部にのみ働くものですので、主軸に選ばれた軸のドライブパルスが設定した速度になるわけではありません。

直線補間、円弧補間は最高8MPPSまで可能です。ただし、連続補間のときは最高4MPPSまでです。ビットパターン補間は4MPPSまで、ヘリカル補間は最高2MPPSまで可能です。

補間ドライブを行うとき、補間速度は必ず設定してください。特に以下のような場合でも、必ず設定が必要です。

- 通常ドライブ後に補間ドライブを行う際、速度パラメータが通常ドライブと同じ場合
- 補間ドライブ後、速度や位置パラメータが同じで補間モードの設定のみを変えて補間ドライブを行う場合

【注意】補間ドライブ中はドライブ速度の変更はできません。

位置データ設定
2・3・4軸直線補間では各軸の終点位置を、円弧補間では円弧の中心位置と終点位置を設定します。2・3・4軸ビットパターン補間では各軸の＋方向/－方向ビットデータを設定します。ビットパターン補間では補間ドライブ開始前に各軸128個のビットデータを書き込むことができます。ヘリカル補間では円弧の中心位置と終点位置、そしてZ・U方向の移動量を設定します。

【注意】位置データが同じ補間ドライブを続けて行う場合でも、位置データは必ず設定してください。

補間ドライブ開始
補間に必要な速度・位置パラメータを設定した後に、補間ドライブ命令を発行すると補間ドライブが開始されます。ビットパターン補間では、補間ドライブ中にビットデータを補充することにより、連続して任意のドライブ軌跡を無限に描くことができます。

補間ドライブ終了待ち
補間ドライブ中は、RR0 （主ステータスレジスタ）の補間を行っているすべての軸のn-DRVビットに1が立ちます。補間ドライブが終了すると当該ビットは0に戻ります。

エラー確認
補間ドライブにおいても、ドライブする各軸のハードリミットエラー、ソフトリミットエラーは作動します。補間ドライブ中、いずれの軸のリミットがアクティブになっても、補間ドライブは停止します。エラーで停止した場合は、RR0（主ステータスレジスタ）の補間指定されている軸のエラービットが1になります。1が立っていれば、その軸のRR2（エラーレジスタ）を読み出し、エラー原因を特定することができます。

【注意】円弧補間、ヘリカル補間、およびビットパターン補間では、＋方向／－方向いずれの方向のハードリミット、およびソフトリミットがアクティブになっても補間が停止する場合があります。従って、円弧補間、ヘリカル補間、およびビットパターン補間によるリミット領域からの脱出はできません。単独で軸をドライブさせて脱出させてください。

補間軸指定クリア
補間ドライブを終了するとき、補間モード設定命令（2Ah）で補間軸の指定を必ずクリアしてください。補間軸を指定したまま通常ドライブを行うと、正常にドライブしない場合があります。

■ サーボモータ用インポジション信号の対応
補間ドライブにおいても、補間ドライブをする各軸のインポジション信号（nINPOS）を有効にすると、補間ドライブ終了後、すべての軸のnINPOS信号がアクティブレベルになるのを待ってから、RR0（主ステータスレジスタ）の補間を行っているすべての軸のドライブビットが0に戻ります。

■ 同期動作による補間ドライブ停止
同期動作を用いて補間ドライブを停止させた場合、必ず補間軸に対しエラー・終了ステータスクリア命令(79h)を発行してください。同期動作によるドライブ終了の確認は、RR2 レジスタの D8 ビットで行うことが可能です。
同期動作については、2.6節を、RR2レジスタに関しては、6.13節を参照してください。

■ nSTOP0, nSTOP1, nSTOP2 信号によるドライブ停止後の補間ドライブ
nSTOP0, nSTOP1, nSTOP2 信号で補間以外のドライブを停止した後、停止した軸を使用して補間ドライブを行う場合、必ず補間軸に対しエラー・終了ステータスクリア命令(79h)を発行してください。ドライブ終了の確認は、RR2 レジスタで行うことが可能です。
RR2レジスタに関しては、6.13節を参照してください。

3.1 直線補間

任意の2軸、3軸または全4軸を選択し、直線補間ドライブを行います。
直線補間は、現在座標に対する終点座標をセットし、補間を行う軸数に応じた直線補間命令を書き込むと実行されます。
終点座標は、現在位置に対する相対値でそれぞれの軸の出力パルス数にセットします。
図3.1-1は2軸補間の例ですが、現在座標から終点座標に向かって、直線補間を行います。図に示すように理想直線に対する位置の演算精度は、全補間範囲内で±0.5 LSBです。

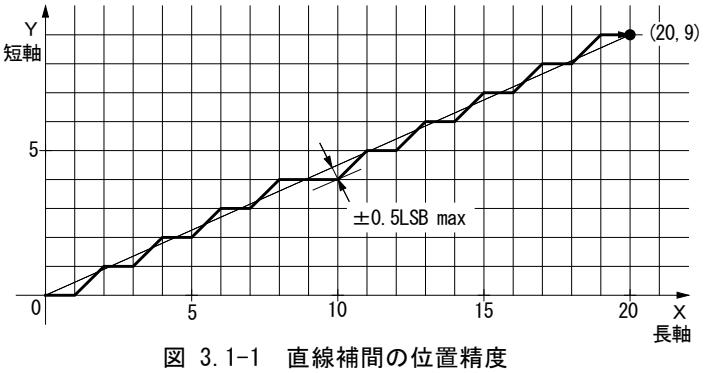


図 3.1-1 直線補間の位置精度

右図3.1-2は、直線補間のドライブパルス出力例です。セットされた終点の値のなかで絶対値が最も大きい軸が長軸となり、補間ドライブ中は、常にパルスを出力します。他の軸は短軸となり、直線補間演算結果により、パルスを出すときと、出さないときがあります。

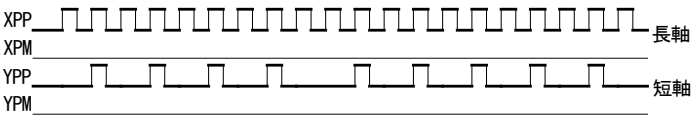


図 3.1-2 終点(X:20,Y:9)のドライブパルス出力例

線速一定モードが無効になっている時には、長軸のドライブパルスの速度が主軸に設定されたドライブ速度となります。
直線補間の座標範囲は符号付き31ビット長です。各軸とも現在位置から－1,073,741,823～＋1,073,741,823の範囲で補間することができます。

3.1.1 終点最大値

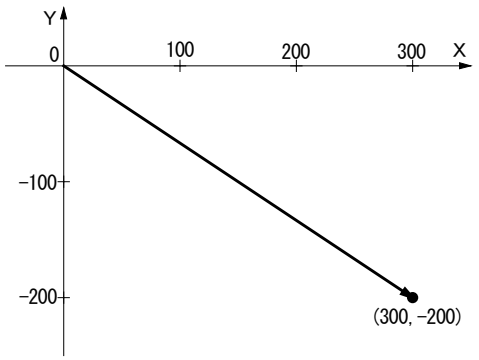
長軸の終点の絶対値を終点最大値と呼びます。
ICのリセット初期状態では終点最大値は自動的に算出されますが、補間モード設定命令(2Ah)でマニュアル設定にすることもできます。マニュアル設定にすると、任意の値を終点最大値として使用することができます。補間モード設定命令(2Ah)については、7.3.8項を参照してください。

3.1.2 直線補間の実例

■ 2軸直線補間ドライブの例

X、Y軸について、現在位置から終点座標(X:+300,Y:-200)まで直線補間します。補間ドライブ速度は、1000PPSの定速ドライブとします。

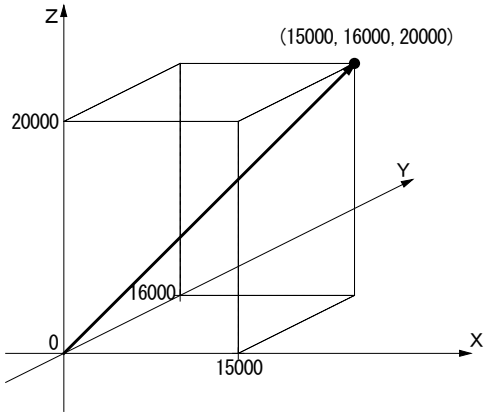
```
WR6 ← 0003h ﾗｲﾄ          ; 補間軸X,Yを指定
WR0 ← 002Ah ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ          ; 初速度：1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ          ; ドライブ速度：1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

WR6 ← 012Ch ﾗｲﾄ          ; 終点Ｘ軸 ：300
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← FF38h ﾗｲﾄ          ; 終点Ｙ軸 ：-200
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR0 ← 0061h ﾗｲﾄ          ; ２軸直線補間ドライブ
```



■ 3軸直線補間ドライブの例

X、Y、Z軸について、現在位置から終点座標(X:15000,Y:16000,Z:20000)まで3軸直線補間します。補間ドライブ速度は、初速度:500PPS、加減速度:40,000PPS/SEC、ドライブ速度:5,000PPSの直線加減速ドライブとします。

```
WR6 ← 0007h ﾗｲﾄ          ; 補間軸X,Y,Zを指定
WR0 ← 002Ah ﾗｲﾄ

WR6 ← 9C40h ﾗｲﾄ          ; 40,000 PPS/SEC
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ          ; 主軸に加速度を設定

WR6 ← 01F4h ﾗｲﾄ          ; 500 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ          ; 主軸に初速度を設定

WR6 ← 1388h ﾗｲﾄ          ; 5000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ          ; 主軸にドライブ速度を設定

WR6 ← 3A98h ﾗｲﾄ          ; 終点Ｘ  : 15,000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 3E80h ﾗｲﾄ          ; 終点Ｙ  : 16,000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR6 ← 4E20h ﾗｲﾄ          ; 終点Ｚ  : 20,000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0406h ﾗｲﾄ

WR0 ← 006Dh ﾗｲﾄ          ; 減速有効
WR0 ← 0062h ﾗｲﾄ          ; ３軸直線補間ドライブ
```



3.2 円弧補間

4軸中、任意の2軸を選択し、円弧補間ドライブを行います。
右図に示す直交座標において、選択した2軸は、X＞Y＞Z＞Uの優先順で優先度の高い軸がax1軸(横軸)に、低い軸がax2軸(縦軸)になります。ax1軸(横軸)は右方向が＋方向に、ax2軸(縦軸)は上方向が＋方向になります。X軸とY軸を選択した場合には、X軸がax1軸(横軸)に、Y軸がax2軸(縦軸)になります。補間モード設定で軸を入れ替えることも可能です。

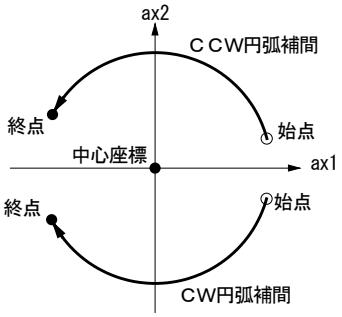


図 3.2-1 ＣＷ／ＣＣW円弧補間

円弧補間は、現在座標(始点)に対する円弧の中心座標、および終点座標をセットし、CW円弧補間命令か、CCW円弧補間命令を書き込むことで実行されます。中心座標、および終点座標の指定は、現在座標(始点)に対する相対値でセットします。
CW円弧補間は、現在座標から、終点座標に向かって、中心座標を中心に時計方向に、また、CCW円弧補間は、反時計方向に円弧を描きます。終点を(0, 0)にすると、真円を描くことができます。
本IC内部の円弧補間の演算では、図3.2-2に示すように、第1軸(ax1)と第2軸(ax2)による平面を、中心座標を中心に、0～7の8つの象限に分けています。図に示すように、0象限では、円弧上を移動する補間座標(ax1, ax2)は、常にax2の絶対値の方がax1の絶対値より小さくなります。絶対値の値が小さい軸の方を短軸とすると、1、2、5、6象限では第1軸(ax1)が短軸になり、0、3、4、7象限では第2軸(ax2)が短軸になります。短軸は、その象限の間、ドライブパルスを常に出力し、長軸は、円弧補間演算結果によって、パルスを出したり出さなかったりします。

図3.2-3は、現在座標から中心(-11, 0)、終点(0, 0)の指定で、半径11の真円を描かせた例です。また、図3.2-4にそのときのドライブパルス出力を示します。

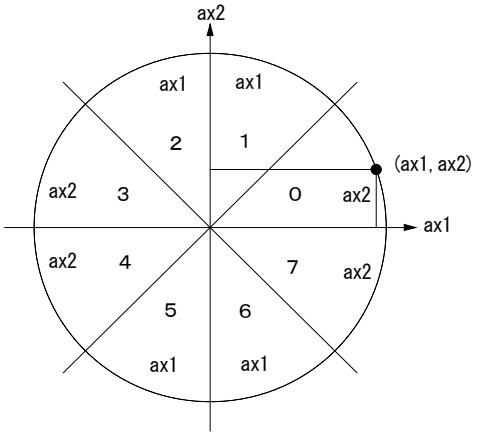


図 3.2-2 円弧補間演算の０～７象限と短軸

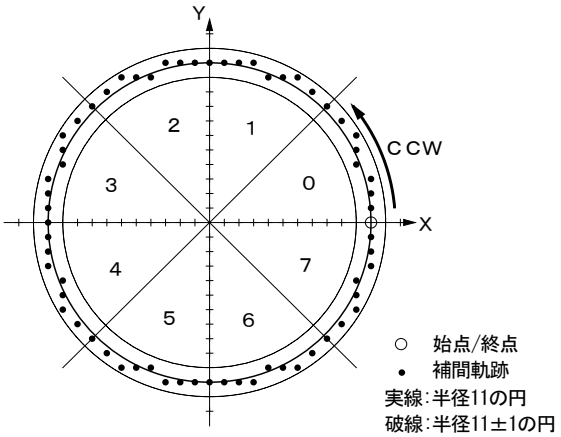


図 3.2-3 円弧補間例

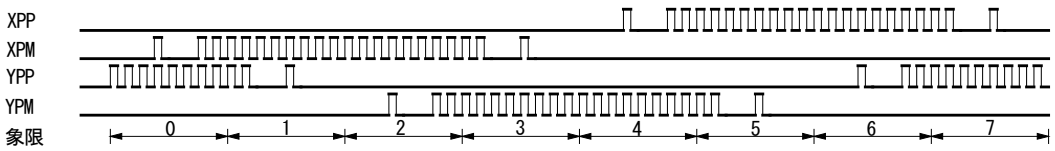


図 3.2-4 円弧補間ドライブパルス出力例

中心座標および終点座標の指定範囲は、現在位置から－1,073,741,823～＋1,073,741,823です。指定円弧曲線に対する位置誤差は全補間範囲内で±1 LSBです。補間速度は1PPS～8MPPSです。

3.2.1 終点判定

円弧補間は、補間ドライブ開始前の現在座標を(0、0)として、中心座標の値によって、半径が決まり、円弧の軌跡を描いていきます。円弧演算の誤差は、補間座標範囲を通じて、±1LSBありますので、指定した終点が必ず円弧の軌跡上にあるとは限りません。そこで、本ICでは、終点のある象限において、終点の短軸の値と等しくなったとき又は越えたときに、円弧補間終了と判断しています。終点のある象限において、終点の短軸の値まで到達できなかった場合には、その象限が終了した所で円弧補間が終了します。
図3. 2-5は、現在位置(0, 0)から中心(-200, 500)、終点(-702, 299)で、CCW円弧補間したときの例です。現在位置(0, 0)と中心(-200, 500)から決まる半径によってCCW方向に補間していきます。指定の終点(-702, 299)は、中心との位置関係から、4象限にあります。補間が4象限に入ると、第2軸(ax2)が短軸となりますので、第2軸の値が終点(-702, 299)の299に達したときに補間終了と判断します。

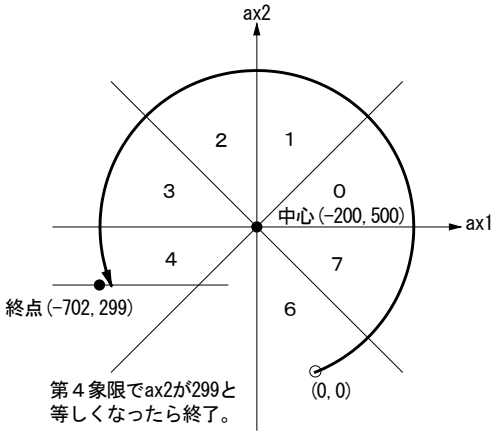


図 3. 2-5 円弧補間終了判定の例

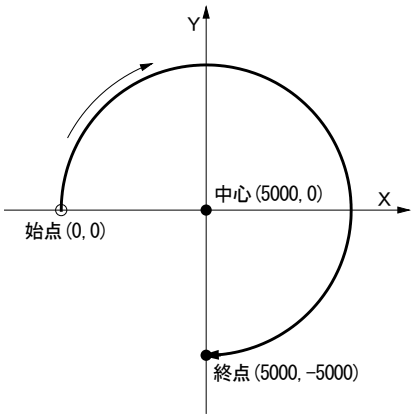
3.2.2 補間軸の入れ替え

円弧補間を行なう軸は、X＞Y＞Z＞Uの優先順で優先度の高い軸がax1(横軸)に、低い軸がax2(縦軸)になります。この軸を入れ替えることが可能です。優先度の低い軸をax1(横軸)、高い軸をax2(縦軸)にしたい場合には、補間モード設定命令(2Ah)でWR6/D4ビットを1にします。

3.2.3 CW円弧補間ドライブの例

X、Y軸について、現在位置(始点)から中心(X:5000, Y:0)、終点(X:5000, Y:-5000)でCW円弧補間します。補間ドライブ速度は、1000PPSの定速ドライブとし、2軸簡易線速一定モードで補間します。

```
WR6 ← 0043h ﾗｲﾄ          ; ax1:Ｘ軸、ax2:Ｙ軸指定、２軸簡易線速一定
WR0 ← 002Ah ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ          ; 主軸に初速度を設定

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ          ; 主軸にドライブ速度を設定

WR6 ← 1388h ﾗｲﾄ          ; 中心Ｘ：5000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ          ; 中心Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← 1388h ﾗｲﾄ          ; 終点Ｘ：5000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← EC78h ﾗｲﾄ          ; 終点Ｙ：-5000
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR0 ← 0064h ﾗｲﾄ          ; ＣＷ円弧補間ドライブ
```



3.3 ヘリカル補間

ヘリカル補間は、XY平面(直交座標)における円弧補間ドライブに同期して他の軸を移動させる動作です。下図はXY平面の円弧補間に合わせて、Z軸を＋方向に移動させている例です。図 3.3-1 は円弧補間 1 周以内のヘリカル補間を示し、図 3.3-2 は複数回転の例を示しています。本 IC ではこれら両方の補間を行わせることができます。

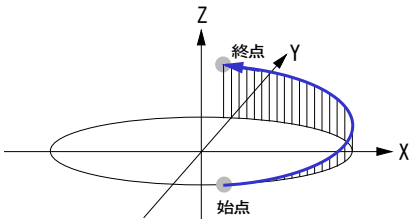


図 3.3-1 ヘリカル補間の例（1 周以内）

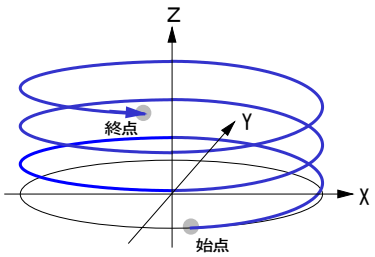


図 3.3-2 ヘリカル補間の例（1 周以上）

また、ヘリカル補間はXY平面上の円弧補間に合わせて、他の回転軸を一定の回転角で回転させ、円弧上の法線動作を行なう事ができます。図3.3-3はXY平面上で円弧補間を行なう台座に回転軸を設けて、台座上のカメラやノズルなどが常に円弧補間の中心を向くように動作する例を表しています。

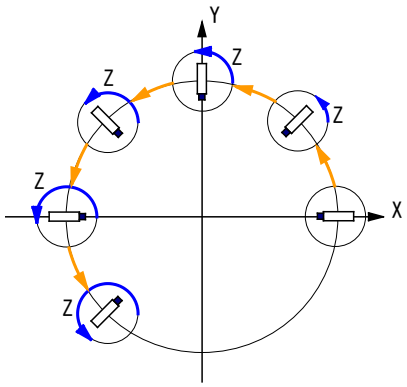


図 3.3-3 ＸＹ軸円弧補間とＺ軸の法線制御の例

本 IC でヘリカル補間を行なうための手順を説明します。ここでは円弧補間とともにZ軸を移動させるヘリカル補間について記述します。本 IC では、Z軸の移動を均一に行なうために円弧補間の総出力パルス数とZ軸の移動パルス数を使用します。Z軸の移動パルス数はあらかじめ決まっているものですが、XY 平面上を動作する円弧補間の総出力パルス数を事前に円弧の中心点、終点の値から精度良く求めることは非常に困難です。そこで本 IC ではヘリカル補間を行なう前に、この円弧補間の総出力パルス数を求めるヘリカル演算を行ないます。実際のヘリカル補間を行うまでの操作順序は次のようになります。

表 3.3-1 ヘリカル補間を行うまでの操作手順

項番	操作項目	内容
#1	補間軸の設定	ヘリカル補間を行う軸を指定する。
#2	補間速度データの設定	円弧補間の速度を設定する。
#3	ヘリカル回転数設定	複数回転させる場合の回転数を設定する。
#4	位置データの設定	円弧補間の中心点と終点を設定する。
#5	ヘリカル演算の実行	円弧補間の総出力パルス数を求める。
#6	位置データの設定	円弧補間の中心点と終点、Z, U軸の送り量を設定する。
#7	ヘリカル補間の実行	ヘリカル補間を実行する。

#1(補間軸の設定)は必ず初めに行なわなければなりません。続いて#2～#4 を行い(#2～#4 の設定順序はありません)、その後#5(ヘリカル演算の実行)を行います。#5(ヘリカル演算の実行)の後、#6(位置データの設定)を行います。円弧補間の中心点と終点については再度設定が必要になります。最後に#7(ヘリカル補間の実行)になります。この順番を変えたり、いずれかの操作項目が無いと正しい補間ができません。

全く同一のヘリカル補間を続けて行う場合においても、#1、#4 および#5 については設定・実行する必要がありませんが、その他の項目についてはすべて再設定・実行しなければなりません。

3.3.1 補間軸の設定

ヘリカル補間の場合には、円弧補間を行わせる軸はXとY軸に固定されています。その他の軸で円弧補間を行わせることはできません。また、円弧補間とともに移動させる軸は、Z軸およびU軸を指定することができます。Z軸とU軸はそれぞれ1軸だけを移動させることも、また両軸をともに移動(あるいは回転)させることも可能です。これにより、例えばカメラやノズル、刃物などをZ軸で円弧補間平面と垂直方向にヘリカル補間させるとともに、U軸によって台座の回転(U軸)を行ないヘッドの法線制御も行なうことができます。

補間を行わせる軸の設定は、補間モード設定命令(2Ah)によって行います。下記のようにWR6レジスタのD0からD3ビットを設定します。補間を行わせる軸に対応するビットを1にします。X軸とY軸のビットは必ず1にし、ZまたはU軸のいずれか、または両方のビットを1にします。

	D15	D14	D13	D12 (H)	D11	D10	D9	D8	D7	D6	D5	D4 (L)	D3	D2	D1	D0
WR6													U-EN	Z-EN	Y-EN	X-EN

D3 U-EN	D2 Z-EN	D1 Y-EN	D0 X-EN	軸動作
0	1	1	1	X,Y軸で円弧補間を行い、円弧補間に合わせてZ軸を移動させる。
1	0	1	1	X,Y軸で円弧補間を行い、円弧補間に合わせてU軸を移動させる。
1	1	1	1	X,Y軸で円弧補間を行い、円弧補間に合わせてZ,U軸を移動させる。

WR6レジスタのその他のビット(D15～D4)も補間に関する機能の設定ビットです。7.3.8項を参照して適正な値を設定する必要があります。

3.3.2 補間速度データの設定

ヘリカル補間の主軸はX軸になりますので、速度設定はX軸に行います。設定できる速度は、1PPS～2MPPS です。通常ヘリカル補間は定速(加減速は行なわない)で行ないますので、X軸に対して初速度とドライブ速度を設定します。この設定された速度でXY平面の円弧補間が動作します。円弧補間に合わせて移動(回転)するZおよびU軸の速度は、円弧補間の速度と軸の送り量によって決まりますので、設定する必要はありません。
補間速度をより一定にさせるために、“短軸パルス均一化モード”と“線速一定モード”が用意されています。これらについては、3.6 節および 3.5 節を参照してください。

3.3.3 ヘリカル回転数の設定

ヘリカル補間を1周以上行なう場合はその回転数を設定します。1周に満たない場合は回転数を0に設定します。WR6レジスタに 0 ～ 65,535 の範囲で回転数を書込み、ヘリカル回転数設定命令(1Ah)を書き込むと回転数が設定されます。このとき、書き込む命令に対して軸指定はする必要ありません。

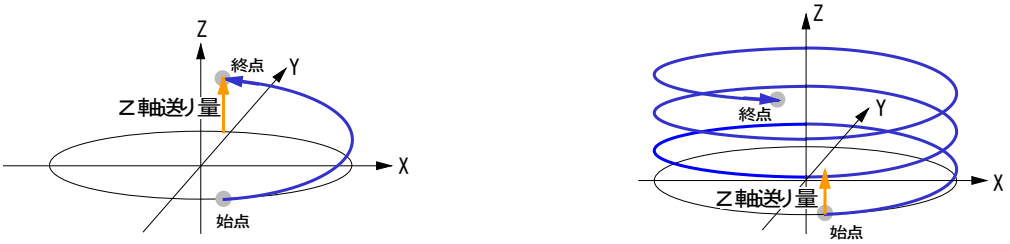
■ 真円のヘリカル補間時の回転数の設定について
終点をX,Y軸共に0を指定すると真円となります。このとき、ヘリカル回転数は0と1のどちらを設定しても1回転となります。2以上を設定すると、設定した数だけ回転します。

3.3.4 位置データの設定

XY平面上を動く円弧補間の中心(X値, Y値)と終点を設定します。さらに円弧補間に合わせてZ軸を送る場合にはZ軸の送り量を、U軸を送る場合にはU軸の送り量を設定します。

表 3.3-2 ヘリカル補間の位置データの設定

設定データ	設定の内容
円弧補間中心	中心点のX軸値, Y軸値を、それぞれ現在位置(ヘリカル補間を始める前の位置)に対する相対パルス数で設定します。WR6,7 レジスタに値を書込み、軸指定とともに円弧中心書き込み命令(08h)を WR0 レジスタに書き込みます。
円弧補間終点	終点のX軸値, Y軸値を、それぞれ現在位置に対する相対パルス数で設定します。WR6,7 レジスタに値を書込み、軸指定とともに移動パルス数／終点書き込み命令(06h)を WR0 レジスタに書き込みます。
Z U軸の送り量	・ 円弧補間に合わせて送る軸の送り量を現在位置に対する相対パルス値で設定します。＋方向に送る場合には正の値で、－方向に送る場合には負の値で設定します。WR6,7 レジスタに値を書込み、軸指定とともに移動パルス数／終点書き込み命令(06h)を WR0 レジスタに書き込みます。 ・ 円弧補間が 1 周に満たない場合には終点までの送り量を設定します（図 3.3-4 (a)参照)。円弧補間が 1 周以上の場合には、円弧補間 1 周分のその軸の送り量を設定します（図 3.3-4 (b)参照)。 ・ 設定するZ軸またはU軸の送り量は、必ず円弧補間の総出力パルス数（ヘリカル演算で得られるヘリカル演算結果の値）より小さな値でなければなりません。目安としては、円弧補間の円弧の長さより小さな値であることが条件になります。



(a) 1 周に満たない場合　　(b) 1周以上の場合

図 3.3-4 Z、U軸の送り量の設定

円弧補間の中心点、終点の設定方法は通常の円弧補間のやり方と同じです。ちょうど1周させる場合には終点を(0,0)に設定します。複数回、回転させて始点の位置で終了させる場合にも終点を(0,0)に設定します。

ヘリカル演算を行うときは、Z軸の送り量、U軸の送り量は設定する必要はありません。

3.3.5 ヘリカル演算の実行

本 IC ではヘリカル補間においてZ軸の移動を均一に行なうために円弧補間の総出力パルス数を予め知っておく必要があります。ヘリカル演算命令はこの総出力パルス数を求めるための命令です。
ヘリカル演算を実行させる前には、補間軸の設定、補間速度データの設定、ヘリカル回転数設定、位置データ(円弧補間の中心点・終点)の設定がなされていなければなりません。ヘリカル演算はこれらのパラメータ値をもとにして実行されます。
ヘリカル演算は、CW ヘリカル演算とCCWヘリカル演算があります。必ずヘリカル補間と同じ円弧回転方向の演算命令を実行させます。回転方向が異なると正しい補間は行えません。
WR0 レジスタにそれぞれ CW ヘリカル演算命令(6Bh)、または CCW ヘリカル演算命令(6Ch)を書き込むと実行されます。

表 3.3-3 ヘリカル演算命令

ヘリカル演算命令コード	実行されるヘリカル演算
6Bh	ＣＷヘリカル演算
6Ch	ＣＣＷヘリカル演算

演算が実行されている間は RR0 レジスタの D0,D1(XDRV、YDRV)ビットが1になり、演算が終了するとこれらのビットが0に戻ります。演算の終了はドライブ終了割り込みを発生させて知る方法もあります。割り込みについては、2.10 節を参照してください。

■ ヘリカル演算結果の読み出しと書き込み
ヘリカル演算を実行し演算が終了すると、ヘリカル演算結果(円弧補間の総出力パルス数)を得ることができます。この値はヘリカル演算値読み出し命令(3Bh)で読み出すことができます。WR0 レジスタにヘリカル演算値読み出し命令(3Bh)を書き込み、RR6,RR7 レジスタから読み出します。
全く同じヘリカル補間(ヘリカル回転数と円弧補間の中心点・終点の値およびZ・U軸送り量がすべて同じ)を何度も行なう時には、補間を行なうたびにヘリカル演算を実行させる必要はありません。1 度得られた演算結果をヘリカル演算値読み出し命令(3Bh)で読み出し、次からはその値を設定するだけでヘリカル補間に移ることができます。ヘリカル演算結果の書き込みは、ヘリカル演算値設定命令(1Bh)で行ないます。WR6,7 レジスタにヘリカル演算結果の値を書き込み、WR0 レジスタにヘリカル演算値設定命令(1Bh)を書き込むと IC 内部レジスタに値が設定されます。

【注意】ヘリカル演算とヘリカル補間は、補間モード設定命令(2Ah)のすべてのビット内容も同じでないと正しく動作しません。

■ ヘリカル演算の実行時間
ヘリカル演算の実行時間を下表に示します。ヘリカル演算の実行時間は、ヘリカル補間時のXY軸円弧の半径によって決まります。演算時間は最大でも円弧 1 周分を演算する時間しかかかりません。ヘリカル回転数が 1 周以上の場合は、何周指定されていても下表の値になります。1 周に満たない場合にはその回転角度に応じて表の値より小さくなります。

表 3.3-4 ヘリカル演算の実行時間

円弧補間の半径 r (パルス)	ヘリカル演算実行時間 t (msec)	
	短軸パルス均一化モード無効	短軸パルス均一化モード有効
1,000	0.7	5.6
10,000	7	56
100,000	70	565
1,000,000	707	5,656

円弧補間の半径は本 IC に設定する円弧中心点(xc,yc)から求めることができます。 円弧半径と実行時間は次式より算出しています。

円弧半径 $r = \sqrt{(xc^2 + yc^2)}$
実行時間 $t = (1 \times 10^{-6} \times r)/\sqrt{(2)}$

【注意】短軸パルス均一化モードでは実行時間は 8 倍されます。

3.3.6 ヘリカル補間の実行

ヘリカル補間を行う前に、3.3.4 項で設定をした位置データをもう一度設定します。その後、CW ヘリカル補間ドライブ命令(69h)または CCW ヘリカル補間ドライブ命令(6Ah)でヘリカル補間が実行されます。XY平面上の円弧補間を CW 回転させる時には CW ヘリカル補間ドライブ命令(69h)、CCW回転させたい時には CCW ヘリカル補間ドライブ命令(6Ah)を WR0 レジスタに書き込むとヘリカル補間が開始されます。

表 3.3-5 ヘリカル補間命令

ヘリカル補間命令コード	実行されるヘリカル補間
69h	ＣＷヘリカル補間
6Ah	ＣＣＷヘリカル補間

ヘリカル補間が実行される前には、すべての必要なデータが設定されていなければなりません。設定項目は 3.3.1 項から 3.3.5 項を参照してください。

3.3.7 現在ヘリカル回転数の読み出し

ヘリカル補間実行中、現在ヘリカル回転数読出し命令(3Ah)で現在の回転数を読み出すことができます。
ヘリカル回転数のカウントアップは、円弧補間が1周して始点に戻ってきたタイミングで行われます。

3.3.8 ヘリカル補間の位置変動

ヘリカル補間は、XY平面上で円弧補間を行い、その円弧補間に同期させて Z 軸またはU軸を移動させます。理想的には円弧補間の中心における回転角の増加量とZ・U軸の送りの増加量は、図 3.3-5 に示すように、正比例しなければなりません。しかし、本 IC の円弧補間はXY直交平面上で行われますので、出力される X 軸とY軸のドライブパルス量の増加は円弧補間の中心における回転角の増加量とは全くの正比例ではありません。円弧補間を行なう X 軸Y軸のドライブパルス量から演算されるZ・U軸の送りもこの影響を受けて完全な正比例ではありません。円弧補間の象限が変わるたびに周期的な変動(ドリフト)が発生します。

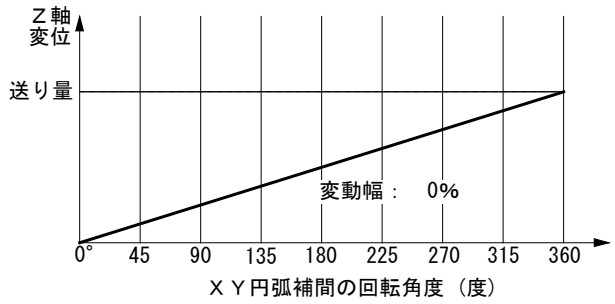


図 3.3-5 理想的なヘリカル補間のＺ軸送り

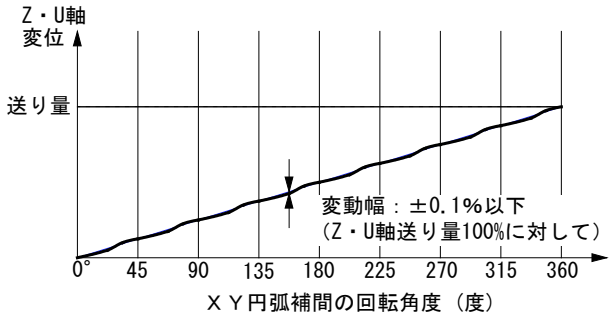


図 3.3-6 MCX514 ヘリカル補間のＺ・Ｕ軸送り変動

本 IC では、図 3.3-6 に示すように、ZまたはU軸の位置は円弧補間の象限が変わるたびに周期的な変動(ドリフト)が発生します。理想位置からの変動幅は、動作条件によって次のようになります。

表 3.3-6 送り量の理想位置からの変動幅

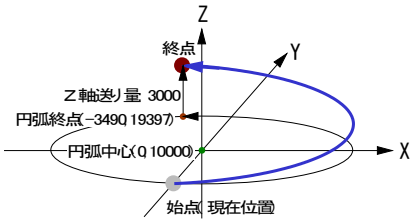
動作条件	理想位置からの変動幅
短軸パルス均一化＋線速一定 2 軸高精度	±0.1%以下
短軸パルス均一化、線速一定ともに無し	±0.4%以下

短軸パルス均一化については、3.6 節を参照してください。
線速一定については、3.5 節を参照してください。

3.3.9　ヘリカル補間の実例

■ 例１ １周未満のヘリカル補間の例（X,Y,Z 軸）

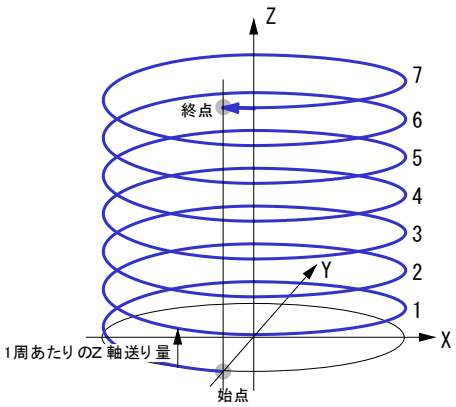
始点(現在位置)から(X:0,Y:10000)の相対位置に中心を持つ CCW 円弧補間を行わせ、終点(X:-3490,Y:19397)で円弧補間を終了させます。このとき、Z 軸を円弧補間の移動に合わせて、現在位置から+3000 移動させます。円弧補間の速度は一定速の 1000PPS とします。




```
WR6 ← 01C7h ﾗｲﾄ          ; ＸＹ軸円弧＋Ｚ軸指定、2軸高精度線速一定
WR0 ← 002Ah ﾗｲﾄ          ; 短軸パルス均一化有効

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ          ; 主軸Ｘに初速度を設定

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ          ; 主軸Ｘにドライブ速度を設定

WR6 ← 0000h ﾗｲﾄ          ; ヘリカル回転数：0
WR0 ← 001Ah ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ          ; 円弧中心Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ          ; 円弧中心Ｙ：10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← F25Eh ﾗｲﾄ          ; 円弧終点Ｘ：-3490
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR0 ← 006Ch ﾗｲﾄ          ; ＣＣＷヘリカル演算（演算時間：約56ms）

RR0 → ﾘｰﾄﾞ               ; 演算終了待ち(D0ビット= 0待ち)

WR6 ← 0000h ﾗｲﾄ          ; 円弧中心Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ          ; 円弧中心Ｙ：10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← F25Eh ﾗｲﾄ          ; 円弧終点Ｘ：-3490
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 4BC5h ﾗｲﾄ          ; 円弧終点Ｙ：19397
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR6 ← 0BB8h ﾗｲﾄ          ; Ｚ送り量：3000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0406h ﾗｲﾄ

WR0 ← 006Ah ﾗｲﾄ          ; ＣＣＷヘリカル補間ドライブ開始

RR0 → ﾘｰﾄﾞ               ; 補間終了待ち(D0ビット= 0待ち)
```

■ 例 2 複数回転のヘリカル補間の例 (X,Y,Z 軸)

始点(現在位置)から(X:0,Y:10000)の相対位置に中心を持つ CW 円弧補間を行わせ、円弧 1 周当たりZ軸を 3000 パルス送り、円弧7周で終了します。円弧補間の速度は一定速の 1000PPS とします。
ヘリカル補間を複数回行わせる時には、Z軸送り量は円弧補間 1 周分の送り量を設定します。




```
WR6 ← 00C7h ﾗｲﾄ          ; ＸＹ軸円弧＋Ｚ軸指定、2軸高精度線速一定
WR0 ← 002Ah ﾗｲﾄ          ; 短軸パルス均一化：無効

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ          ; 主軸Ｘに初速度を設定

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ          ; 主軸Ｘにドライブ速度を設定

WR6 ← 0007h ﾗｲﾄ          ; ヘリカル回転数：７
WR0 ← 001Ah ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ          ; 円弧中心Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ          ; 円弧中心Ｙ：10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← F25Eh ﾗｲﾄ          ; 円弧終点Ｘ：0
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 4BC5h ﾗｲﾄ          ; 円弧終点Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR0 ← 006Bh ﾗｲﾄ          ; ＣＷヘリカル演算（演算時間：約56ms）

RR0 → ﾘｰﾄﾞ               ; 演算終了待ち(D0ビット= 0待ち)

WR6 ← 0000h ﾗｲﾄ          ; 円弧中心Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ          ; 円弧中心Ｙ：10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← F25Eh ﾗｲﾄ          ; 円弧終点Ｘ：0
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 4BC5h ﾗｲﾄ          ; 円弧終点Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR6 ← 0BB8h ﾗｲﾄ          ; Ｚ送り量：3000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0406h ﾗｲﾄ

WR0 ← 0069h ﾗｲﾄ          ; ＣＷヘリカル補間ドライブ開始

RR0 → ﾘｰﾄﾞ               ; 補間終了待ち(D0ビット= 0待ち)
```

■ 例 3　Z,U 両軸によるヘリカル補間の例 (X,Y,Z 軸)

半径 10000 の円弧補間を CCW 方向に1周させます。円弧補間 1 周の間に、Z軸を 3000 パルス送り、回転軸であるU軸を1回転(400 パルス)させます。円弧補間の速度は一定速の 1000PPS とします。

```
WR6 ← 00CFh ﾗｲﾄ          ; ＸＹ軸円弧＋Ｚ,Ｕ軸指定、2軸高精度線速一定
WR0 ← 002Ah ﾗｲﾄ          ; 短軸パルス均一化：無効

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ          ; 主軸Ｘに初速度を設定

WR6 ← 03E8h ﾗｲﾄ          ; 1000 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ          ; 主軸Ｘにドライブ速度を設定

WR6 ← 00007h ﾗｲﾄ         ; ヘリカル回転数：1
WR0 ← 001Ah ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ          ; 円弧中心Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ          ; 円弧中心Ｙ：10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← F25Eh ﾗｲﾄ          ; 円弧終点Ｘ：0
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 4BC5h ﾗｲﾄ          ; 円弧終点Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR0 ← 006Ch ﾗｲﾄ          ; ＣＣＷヘリカル演算 (演算時間：約56ms)

RR0 → ﾘｰﾄﾞ               ; 演算終了待ち(D0ビット= 0待ち)

WR6 ← 0000h ﾗｲﾄ          ; 円弧中心Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 2710h ﾗｲﾄ          ; 円弧中心Ｙ：10000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← F25Eh ﾗｲﾄ          ; 円弧終点Ｘ：0
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 4BC5h ﾗｲﾄ          ; 円弧終点Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR6 ← 0BB8h ﾗｲﾄ          ; Ｚ軸送り量：3000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0406h ﾗｲﾄ

WR6 ← 0190h ﾗｲﾄ          ; Ｕ軸送り量：400
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0806h ﾗｲﾄ

WR0 ← 006Ah ﾗｲﾄ          ; ＣＣＷヘリカル補間ドライブ開始

RR0 → ﾘｰﾄﾞ               ; 補間終了待ち(D0ビット= 0待ち)
```

3.4 ビットパターン補間

本 IC のビットパターン補間は、1 ドライブパルス単位で＋方向または－方向パルスを出力するか否かを指定して複数軸の補間を行う動作です。２軸から最大４軸までを補間することができます。

CPUから補間を行う各軸に対して、＋方向または－方向のドライブパルスを1ビット1パルスでセットします。ドライブパルスを出すときは"1"、出さないときは"0"にセットします。例えば、右図3.4-1のような軌跡を描く場合、X＋方向、X－方向、Y＋方向、Y－方向のそれぞれのドライブパルスを出すときは"1"、出さないときは"0"とすると、ビットパターンデータは、下のようになります。

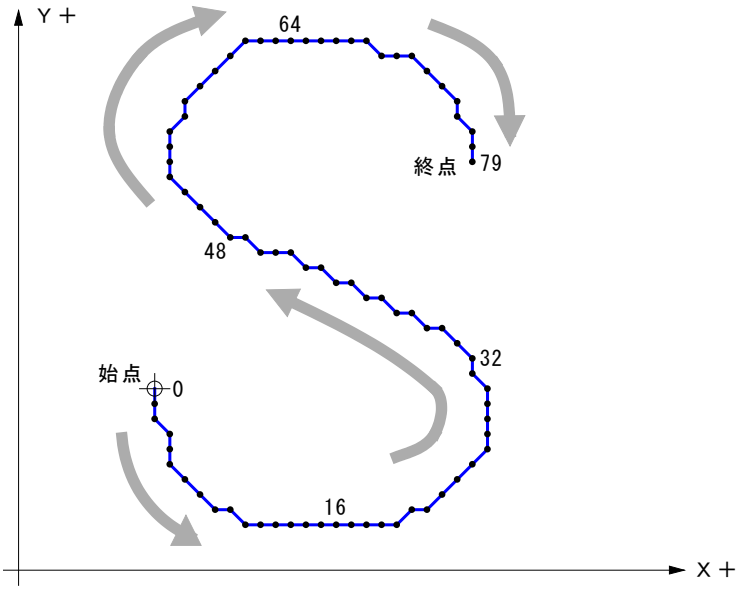


図 3.4-1 ＸＹ軸のビットパターン補間例

```
79              64              48              32              16              0
10010111 11111111 11111110 10000000 00000000 00000000 00000011 11111111 11111111 11100100 :XPP(X＋方向パルス)
10000000 00000000 00000000 00001111 11111111 11111111 01000000 00000000 00000000 00000000 :XPM(X－方向パルス)
00000000 00000000 00011111 11111111 01001010 10101011 11111111 11010000 00000000 00000000 :YPP(Y＋方向パルス)
01111111 00100000 00000000 00000000 00000000 00000000 00000000 00000000 00000011 11111111 :YPP(Y－方向パルス)
```

ビットパターン補間動作を行うための操作手順は次のようになります。

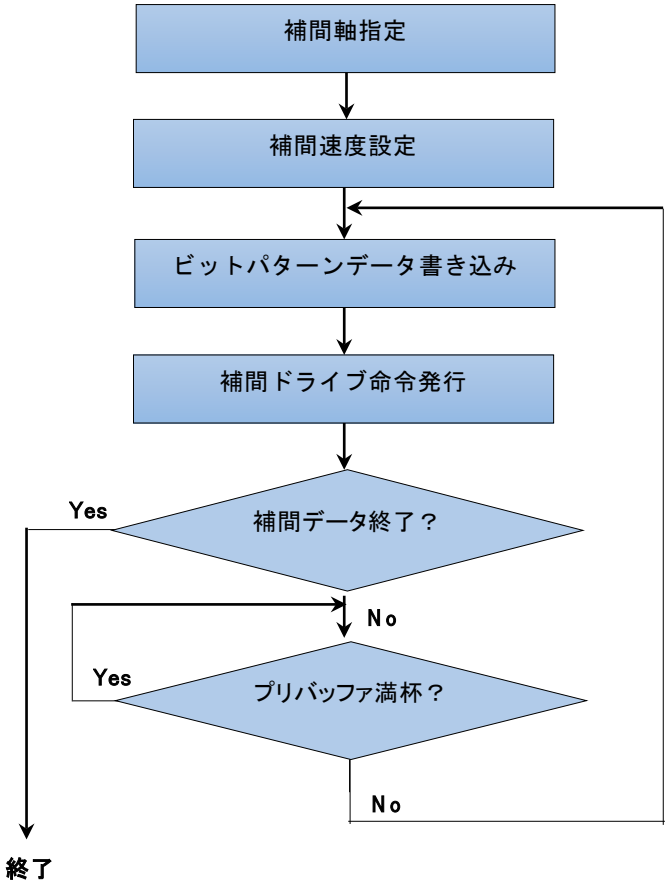


図 3.4-2 ビットパターン補間操作手順

3.4.1 補間軸指定

補間を行わせる軸の設定は、補間モード設定命令(2Ah)によって行います。下記のようにWR6レジスタのD0からD3ビットを設定します。補間を行わせる軸に対応するビットを1にします。2軸から4軸全軸をビットパターン補間させることが可能です。1軸のみを指定することはできません。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6													U-EN	Z-EN	Y-EN	X-EN

WR6レジスタのその他のビット(D15～D4)も補間に関する機能の設定ビットです。7.3.8項を参照して適正な値を設定する必要があります。

3.4.2 補間速度設定

補間を行う軸の中の主軸に対して、ビットパターン補間を行うドライブの速度を設定します。
BP(ビットパターン)補間のドライブ速度は、最高4MHzまで可能です。しかし、ビット数が128ビットを越える場合には、CPUは補間ドライブ中にBPデータを後述のプリバッファに補充していかなければなりませんので、補間ドライブ速度は、CPUのBPデータのセットアップに要する時間に依存することになります。
例えば2軸ビットパターン補間を行う場合、CPU側はBPデータをセットするのに、(16ビットデータ×2＋16ビット命令)×2軸＋補間ドライブ命令を書き込まなければなりません。これに要する時間が100μSECかかるとすると、16ビット(=16ドライブパルス)を出力する時間はこれより長くなければいけません。従って、補間ドライブ速度は 1/(100μSEC/16)=160KPPS以下の速度でなければならないことになります。これより速い速度にするとBPデータの補充が間に合わなくなります。

3.4.3 ビットパターンデータの書き込み

補間を行う軸それぞれのビットパターンデータを書き込みます。
WR6レジスタに16ビット分の＋方向のビットデータを書き込み、WR7レジスタに16ビット分の－方向のビットデータを書き込みます。16ビットデータはD0ビットから上位ビットに向かって順にドライブパルスとして出力されます。
軸指定とともにBPデータ設定命令(06h)を WR0 レジスタに書き込むとBPデータがプリバッファに格納されます。これを、補間を行うすべての軸について行います。

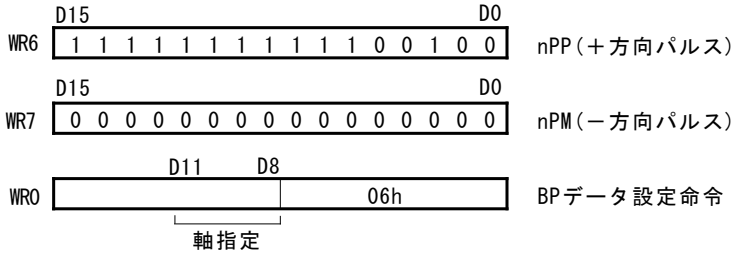


図 3.4-3 ビットパターンデータの書き込み

3.4.4 補間ドライブ命令の発行

すべての軸のビットパターンデータを書き込み終えたら、ビットパターン補間ドライブ命令を WR0 レジスタに書き込みます。
2軸から4軸までを補間することができ、補間ドライブ命令のコードはそれぞれ下記のようになります。

表 3.4-1 ビットパターン補間命令

補間命令	命令コード
２軸ビットパターン補間	66h
３軸ビットパターン補間	67h
４軸ビットパターン補間	68h

軸指定は不要です。命令を WR0 レジスタに書き込むとプリバッファが1段更新されて(スタックカウンタが1つ増加される)、補間ドライブが直ちに実行されます。プリバッファにある程度ビットパターンデータを蓄えてから補間を開始させたい場合には、事前に主軸に対してドライブ開始ホールド命令(77h)をセットしておきます。ビットパターンデータと補間命令を複数段書き込んだあとドライブ開始フリー命令(78h)を主軸に対して発行すると補間ドライブが開始されます。

【注意】全軸のビットパターンデータ書き込みのあと必ずビットパターン補間命令を書き込む必要があります。プリバッファの更新はビットパターン補間命令の書き込みによって行われます。

3.4.5 補間終了

ビットパターン補間は、次の2通りの方法で終了させることができます。

① 補間軸のビットパターンデータに終了コードを書き込む。
ビットパターン補間軸のいずれかの軸の＋方向、－方向のビットデータをともに"1"にすると、ビットパターン補間終了と判断します。終了コード以降にビットパターンデータが書かれてあってもすべて無効になります。

図 3.4-4 終了コードによるビットパターン補間終了

② データ書き込みを中止する。
ビットパターンデータの書き込みを中止すると、プリバッファにスタックされているすべてのビットパターンデータをドライブパルスとして吐き出したのち、補間ドライブを終了します。

3.4.6 プリバッファの空きを確認

本ICは連続補間ドライブ用に8段プリバッファが用意されています。ビットパターン補間の場合には、補間を行う全軸とも16ビットのパターンデータを8段、すなわち16×8＝128ビットを蓄えることができます。128ビットを超える補間を行う場合には、プリバッファの空きを確認しながら補間を行う必要があります。RR0 レジスタの D12～D15 の 4 ビットは、このプリバッファのスタックカウンタ値の表示に割り当てられています。4 ビットの値が0の時は完全な空状態を示しています。8になると満杯の状態を示していますので、それ以上BPデータを書き込むことはできません。ビットパターン補間命令を書き込むとスタックカウンタ値は1つ増加すると同時に補間ドライブが始まります。16 ビットを出力し終えるとスタックカウンタ値は1つ減少します。
また、RR0 レジスタの D11(CNEXT)ビットは連続補間ドライブの次データの書込み可能を知らせるビットです。補間ドライブが開始されると、プリバッファのスタックカウンタが1から7の間は CNEXT ビットが1になります。上位 CPU はこのビットが1であれば、次データを書き込んでも良いと判断できます。

3.4.7 補間ドライブの中断

■ 停止命令による中断

ビットパターン補間ドライブを行っている主軸に対して、即停止命令、あるいは減速停止命令を書き込むと、補間ドライブは停止します。
プリバッファのスタックカウンタは強制的に0になります。プリバッファにスタックされていたビットパターンデータはすべて無効となります。

■ ハードリミット、ソフトリミットによる中断

補間ドライブ中は、いずれの軸のハードリミット、ソフトリミットがアクティブになっても、補間ドライブは停止します。
ビットパターン補間では、＋方向／－方向いずれの方向のハードリミット、およびソフトリミットがアクティブになっても補間が停止する場合があります。従って、ビットパターン補間によるリミットオーバ領域からの脱出はできませんので、ご注意ください。

3.4.8 ビットパターン補間の実例

X軸とY軸の m×16ビットのビットパターン補間を行います。図3.4-1のビットパターン図形を例にすると、79ビットありますので m＝5 になります。補間ドライブ速度は1000PPSの定速ドライブとし、2軸簡易線速一定モードで補間します。X軸とY軸の補間では主軸がX軸になりますので、ドライブ速度はX軸に設定します。

ビットパターンデータはメモリ上に次のように格納しておきます。

	m = 5	m = 4	m = 3	m = 2	m = 1
X軸＋方向データ X_PlusBPdata(m)	1001011111111111 97FFh	1111111010000000 FE80h	0000000000000000 0000h	0000001111111111 03FFh	1111111111100100 FFE4h
X軸－方向データ X_MinusBPdata(m)	1000000000000000 8000h	0000000000001111 000Fh	1111111111111111 FFFFh	0100000000000000 4000h	0000000000000000 0000h
Y軸＋方向データ Y_PlusBPdata(m)	0000000000000000 0000h	0001111111111111 1FFFh	0100101010101011 4AABh	1111111111010000 FFD0h	0000000000000000 0000h
Y軸－方向データ Y_MinusBPdata(m)	0111111100100000 7F20h	0000000000000000 0000h	0000000000000000 0000h	0000000000000000 0000h	0000001111111111 03FFh

```
    WR6 ← 0043h ﾗｲﾄ                    ; Ｘ軸とＹ軸指定、２軸簡易線速一定
    WR0 ← 002Ah ﾗｲﾄ                    ; 補間モード設定

    WR6 ← 03E8h ﾗｲﾄ                    ; Ｘ軸（主軸）速度パラメータ設定
    WR7 ← 0000h ﾗｲﾄ                    ; 初速度：1000 PPS
    WR0 ← 0104h ﾗｲﾄ

    WR6 ← 03E8h ﾗｲﾄ                    ; ドライブ速度：1000 PPS
    WR7 ← 0000h ﾗｲﾄ
    WR0 ← 0105h ﾗｲﾄ

    m   ← 1                            ; データポインタ＝１

Loop:

    WR6 ← X_PlusBPdata(m)  ﾗｲﾄ         ; Ｘ軸＋方向BPデータ
    WR7 ← X_MinusBPdata(m) ﾗｲﾄ         ; Ｘ軸－方向BPデータ
    WR0 ← 0106h ﾗｲﾄ

    WR6 ← Y_PlusBPdata(m)  ﾗｲﾄ         ; Ｙ軸＋方向BPデータ
    WR7 ← Y_MinusBPdata(m) ﾗｲﾄ         ; Ｙ軸－方向BPデータ
    WR0 ← 0206h ﾗｲﾄ

    WR0 ← 0066h ﾗｲﾄ                    ; ２軸ＢＰ補間命令     このステップの初めの実行で補間ドライブが開始される。

    m   ← m + 1                        ; データポインタ増加
    m = 6 ならば終了

    RR0 → ﾘｰﾄﾞ                         ; プリバッファの空きを確認
    RR0/D11 = 1 ならば Loop へジャンプ。= 0ならばRR0ﾘｰﾄﾞへ
```

■ 割り込みを用いたビットパターン補間ドライブ

連続補間専用の割り込み信号(INT1N)が用意されています。この信号は、プリバッファのスタックカウンタが8から7に変化した時、または4から3に変化した時にアクティブ(Lowレベル)になります。
上位CPUは、この割り込み信号が発生したら、スタックカウンタが8になるまで(＝ CNEXTビットが1の間)次のBPデータを書き込むことができます。すなわち、8から7を選択時には1段、4から3を選択時には5段連続して次のBPデータを書き込むことができます。

割り込み信号(INT1N)は、ビットパターンデータを書き込んだあと、補間命令(2/3/4軸ビットパターン補間命令など)を書き込むと非アクティブに戻ります。また、補間ドライブが終了すると強制的に非アクティブに戻ります。

3.5 線速一定

線速とは、補間ドライブを行なう時の軌跡先端の移動速度のことで、ヘッドスピードとも言います。補間ドライブを行いながらワークの加工を行なう、塗布するなどの動作では、この線速を一定に保つことが重要な条件になります。本 IC では2軸補間においては、2軸簡易線速一定モードと2軸高精度線速一定モードが用意されています。また、3軸補間では3軸簡易線速一定モードが用意されています。

図3.5-1は、XY直行平面における2軸補間の軌跡を示しています。主軸からの基本パルスに従って各軸がドライブパルスを出力していきますが、図に示すように、X, Y軸両方ともドライブパルスが出力されるときは、1軸だけのドライブパルス出力に比べて、1.414倍長い距離を移動することになります。
線速一定の機能を働かせないと、両軸ともドライブパルスを出力する時は移動距離が1.414倍長い距離にもかかわらず同じ周期でドライブパルスを出力しますので、速度が1.414倍速くなることになります。

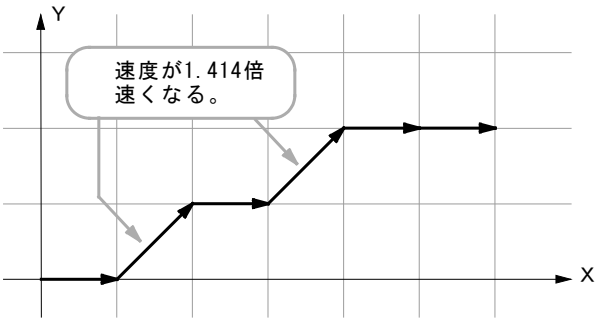
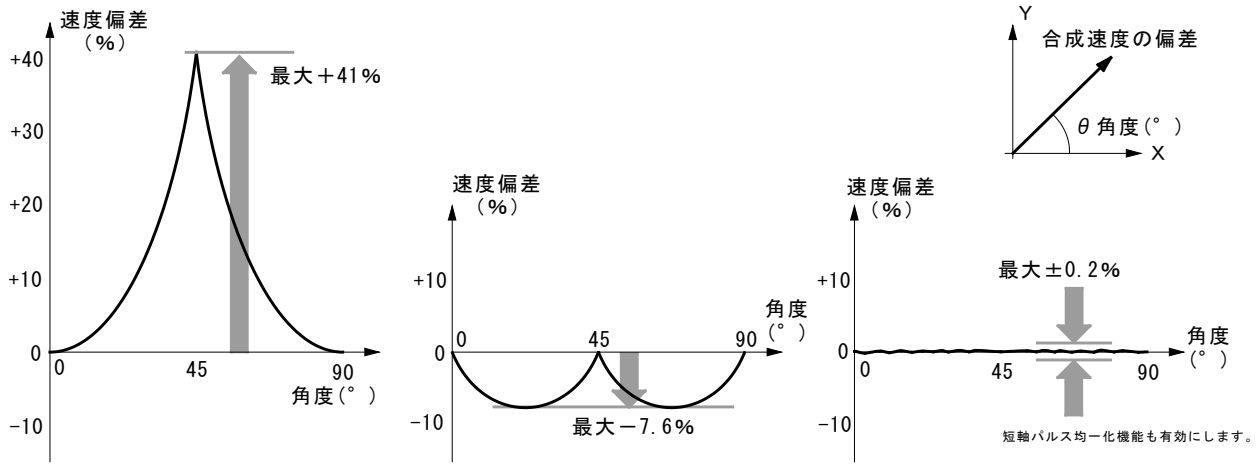


図 3.5-1 ２軸補間例

下図3.5-2は、XY直行平面において直線補間を行うとき、X軸線と補間する直線のなす角度が0から90度の範囲での線速度の偏差を表しています。図は0-90度の範囲を記載していますが、90－180、180－270、270－360 の範囲も同様です。



a． 線速一定：無効　　b． ２軸簡易線速一定　　c． ２軸高精度線速一定

図 3.5-2 直線補間ドライブの設定速度に対する線速度の偏差

図3.5-2 aは、線速一定機能を無効にしたときの設定ドライブ速度に対する線速度の偏差を表しています。X軸からの角度が45度のとき速度偏差が最大になり、約＋41%速度が増加します。
図3.5-2 bは、2軸簡易線速一定モードでの速度偏差を示しています。2軸簡易線速一定モードでは、両軸ともドライブパルスが出力されるときのパルス周期を1／1.414倍にして速度偏差を改善する方法です。
図3.5-2 cは、2軸高精度線速一定にモード設定したときの速度偏差を示しています。すべての角度範囲において、速度偏差は±0.2%以下[*1]に抑えることができます。*1：短軸パルス均一化機能を有効にする必要があります。

3軸直線補間では3軸簡易線速一定モードが用意されています。3軸簡易線速一定モードでは、3軸中いずれかの2軸のドライブパルスが出力される時にはパルス周期が1／1.414倍にし、3軸すべてのドライブパルスが出力される時にはパルス周期が1／1.732倍にして速度偏差を改善します。

3.5.1 線速一定の設定

線速一定は補間モード設定(2Ah)の D6,D7 ビットの 2 ビットで設定します。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6									SPD1	SPD0						

それぞれの線速一定モードに対応するD6,D7ビットの設定は下表のようになります。

表 3.5-1 線速一定モードの設定

D7(SPD1)ビット	D6(SPD0)ビット	線速一定モード
0	0	無効
0	1	2 軸簡易線速一定
1	0	3 軸簡易線速一定
1	1	2 軸高精度線速一定

■ 高精度2軸線速一定の直線補間ドライブ例

X軸とY軸の直線補間をドライブ速度 1000PPS の定速ドライブ、2軸高精度線速一定モードで行う例です。短軸パルス均一化機能も有効にしています。

```
// 補間モード設定
WR6 ← 01C3h ﾗｲﾄ          // X,Y軸補間、２軸高精度線速一定モード、短軸パルス均一化有効
WR0 ← 002Ah ﾗｲﾄ          // 補間モード設定命令

// 主軸のドライブ設定
WR6 ← 03E8h ﾗｲﾄ          // 初速度：1000pps
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 03E8h ﾗｲﾄ          // ドライブ速度：1000pps
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

// 終点設定
WR6 ← 03E8h ﾗｲﾄ          // 終点 X値：1000
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 0190h ﾗｲﾄ          // 終点 Y値：400
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

// 補間ドライブ開始
WR0 ← 0061h ﾗｲﾄ          // 2軸直線補間ドライブ
```

3.6 短軸パルス均一化

通常の補間ドライブでは、補間を行なうすべての軸がドライブ中、常に均一の周期でドライブパルスを出力するわけではありません。例えば下図 3.6-1 a. に示すように、2 軸直線補間において、移動量(パルス)が多い方の軸(長軸)は常にパルスを出力し続けますが、少ない方の軸(短軸)は補間演算結果によりパルスを出力するときとしないときがあります。ステッピングモータの場合には、この短軸のパルスの間引きにより機械的振動が増大する場合があります。

短軸パルス均一化はこの問題を改善する機能です。移動量が少ない軸においても、極力パルス周期を均一に近づけてドライブパルスを出力します。下図3.6-1 b. は、短軸パルス均一化機能を有効にしたときの出力ドライブパルスの波形を示しています。

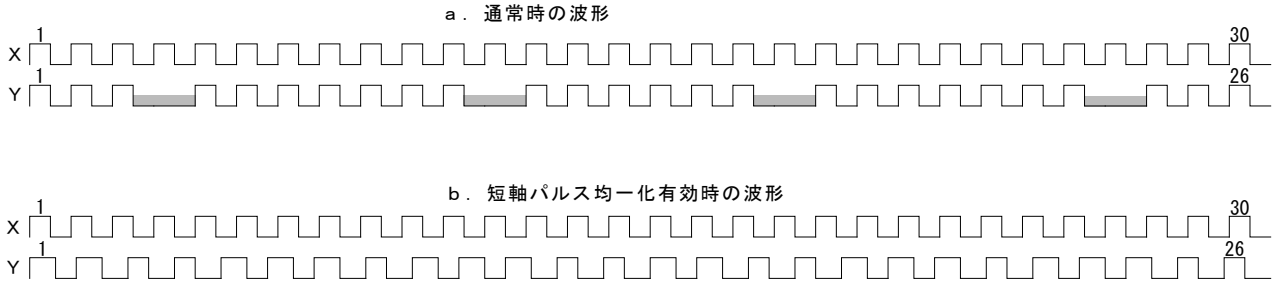


図 3.6-1 ２軸直線補間（終点 X30, Y26）のパルス波形

短軸パルス均一化は、IC の内部の補間演算を通常より数倍高めて行っています。そのため、設定できるパラメータの設定範囲は下表にように 1/8 に制限されます。 短軸パルス均一化を有効にする場合は、必ず下表に示す設定可能範囲内で補間ドライブを行ってください。

表 3.6-1 短軸パルス均一化のパラメータ設定範囲

設定パラメータ	記号	設定可能範囲	
		短軸パルス均一化：有効	通常
ドライブ速度	V	1～1,000,000	1～8,000,000
初速度	SV	1～1,000,000	1～8,000,000
加速度,減速度	A, D	1～67,108,863	1～536,870,911
終点	P	-134,217,728～+134,217,728	-1,073,741,823～+1,073,741,823
円弧中心点	C	-134,217,728～+134,217,728	-1,073,741,823～+1,073,741,823

3.6.1 短軸パルス均一化の設定

短軸パルス均一化は補間モード設定命令(2Ah)のD8ビットで設定します。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6								LMDF								

1を設定すると短軸パルス均一化が有効になり、0 を設定すると短軸パルス均一化が無効になります。

3.6.2 短軸パルス均一化使用時の注意点

- 次の補間ドライブでは短軸パルス均一化は使用できません。
  - ① S 加減速ドライブ
  - ② マルチチップ補間
  - ③ 補間ステップ送り
  - ④ BP 補間
  - ⑤ 連続補間ドライブ
  - ⑥ 多目的レジスタを用いた現在ドライブ速度の比較動作
  - ⑦ 現在ドライブ速度と現在加減速度を多目的レジスタにセットする同期動作

- 円弧補間、ヘリカル補間で短軸パルス均一化を使用した場合、円弧の始点と終点がＸ軸またはＹ軸上以外の点となった場合、終点位置が両軸ともに±1 パルスずれることがあります。ヘリカル補間ではこのずれが累積していくことがあります。円弧補間、ヘリカル補間で円弧の始点と終点が X 軸または Y 軸上以外の点となった場合に短軸パルス均一化を使用するときは、このずれが問題になるか否かを十分ご検討ください。円弧の始点と終点がともに X 軸または Y 軸の軸上にある場合は、このずれは発生しません。

3.7 連続補間

連続補間は、直線補間→円弧補間→直線補間→…というように、各々の補間セグメントを、ドライブを停止しないで、連続して補間を行う動作です。連続補間を行う軸数が同じであることが前提であり、下表のような連続補間が可能です。

表 3.7-1 実行可能な連続補間

実行可能な連続補間	動 作
2軸直線補間の連続	2軸直線 → 2軸直線 → 2軸直線 → ・・・・・・
2軸直線補間と円弧補間の連続	2軸直線 → 円弧補間 → 2軸直線 →2軸直線 → 円弧補間 → ・・・・
3軸直線補間の連続	3軸直線 → 3軸直線 →3軸直線 → ・・・・・・
4軸直線補間の連続	4軸直線 → 4軸直線 →4軸直線 → ・・・・・・

連続補間はプリバッファを用いて実現します。補間ドライブ開始前、あるいは実行中に、プリバッファに補間データをセットする事で連続した補間ドライブを実現します。プリバッファには最大8セグメント分の補間データをセットする事ができます。

3.7.1 連続補間の実施方法

連続補間を行う場合、プリバッファにあらかじめ補間データをセットしてから補間ドライブを開始します。補間ドライブ開始前に最大 8 セグメント分の補間データをプリバッファにセット可能です。補間ドライブ開始後は、プリバッファのスタックカウンタ値を確認しながら、次の補間データ(セグメントデータ)をセットすることで連続補間を実現します。

連続補間は、次の手順で行います。

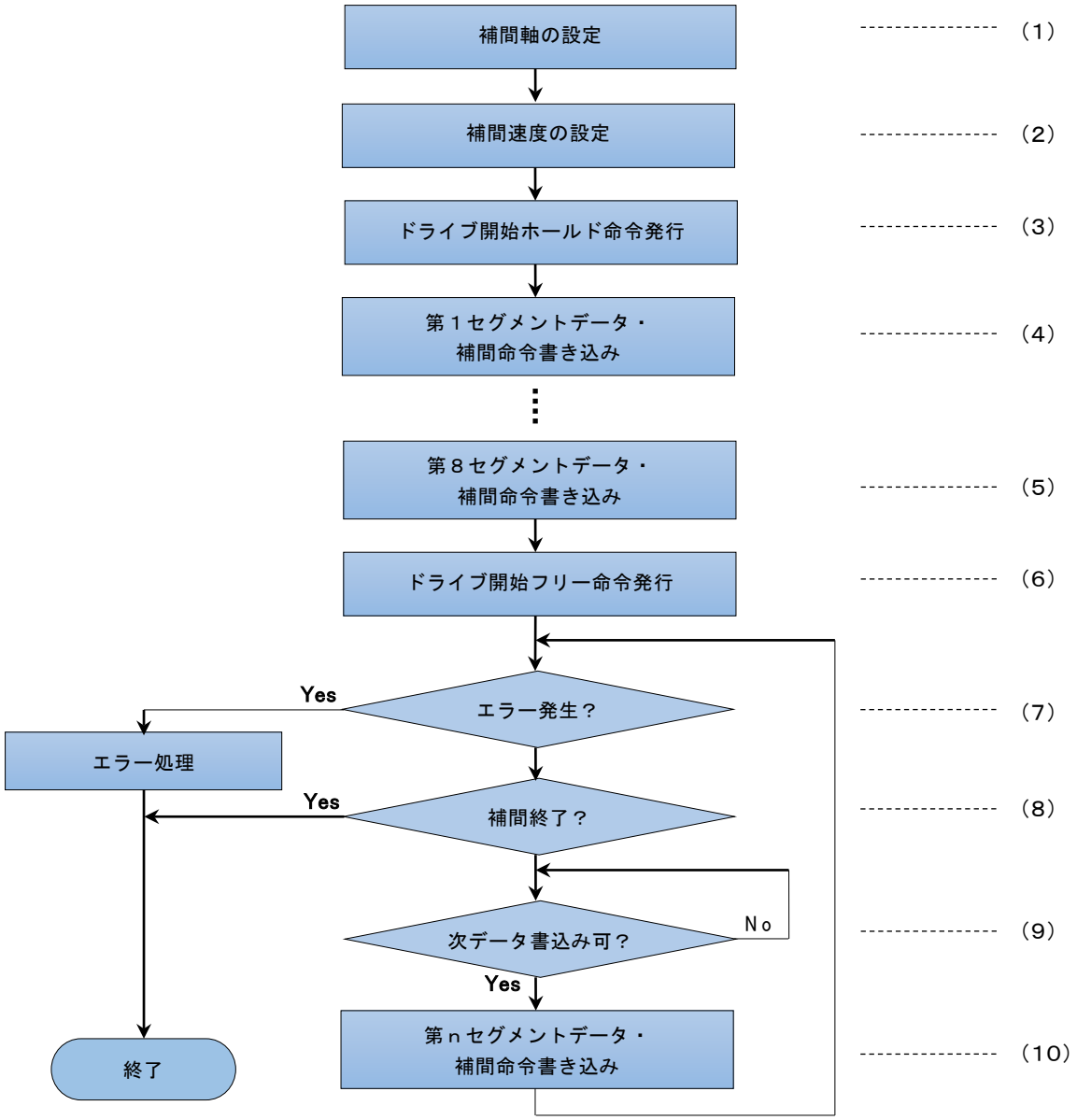


図 3.7-1 連続補間の流れ

(1) 補間軸の設定

補間モード設定命令(2Ah)によって、補間軸を設定します。下記のように WR6 レジスタの D0 から D3 ビットを設定します。補間を行わせる軸に対応するビットを1にします。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6													U-EN	Z-EN	Y-EN	X-EN

・ 連続補間のドライブ開始後は補間軸設定の変更はできません。
・ WR6レジスタのその他のビット(D15～D4)も補間に関する機能の設定ビットです。7.3.8項を参照して適正な値を設定する必要があります。

(2) 補間速度設定

補間を行う軸の中の主軸に対して、補間を行うドライブの速度を設定します。設定できる最高速度は4MPPS までです。
連続補間をすべてのセグメントを通して定速ドライブで行う場合には、初速度をドライブ速度と同じ値に設定します。

(3) ドライブ開始ホールド命令の発行

主軸に対してドライブ開始ホールド命令(77h)を発行します。ドライブ開始ホールド命令を発行すると、そのあと補間ドライブ命令を発行してもドライブが開始されなくなります。これにより、プリバッファに補間開始前に最大 8 セグメント分の補間データをセットする事が可能になります。

(4) 第1セグメントのデータ・補間命令書き込み

第1セグメントが直線補間の場合には、各補間軸に対して終点を書き込み、その後直線補間ドライブ命令を書き込みます。円弧補間の場合には各軸に対して円弧中心点および終点を書き込み、その後円弧補間命令を書き込みます。
1つのセグメント情報の書き込みにおいては、終点、円弧中心点、補間軸の書き込みの順番はどれが先でも構いませんが、補間ドライブ命令は必ず最後に書き込む必要があります。

(5) 第8セグメントまでのデータ・補間命令書き込み

第2セグメントから第8セグメントまで、第1セグメントと同様に、データと補間ドライブ命令を書き込んでいきます。
プリバッファは、8段構成になっています。RR0レジスタのD12～D15に示されるスタックカウンタ値を確認しながら、補間ドライブを開始する前に8セグメントまで書き込むことができます。

(6) ドライブ開始フリー命令の発行

プリバッファに必要なセグメント分の補間データを書き込んだ後、主軸に対してドライブ開始フリー命令(78h)を発行します。このタイミングで、補間ドライブが開始されます。

(7) エラーの確認

RR0 レジスタの D4～D7(X～UERR)ビットはドライブを行っている軸のエラー発生の有無が示されます。エラーが発生すると該当ビットに1が立ち、補間ドライブは停止します。これらのビットを確認し、エラーが発生していなければ次の処理を行います。
RR0 レジスタのエラービットの詳細は 6.13 節を参照してください。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
RR0	HSTC3	HSTC2	HSTC1	HSTC0	CNEXT	ZONE2	ZONE1	ZONE0	U-ERR	Z-ERR	Y-ERR	X-ERR	U-DRV	Z-DRV	Y-DRV	X-DRV

(8) 補間終了の確認

すべてのセグメントの書込みが終了か否かを確認し、終了でなければ次の処理を行います。

(9) 次データ書き込み可を確認

RR0 レジスタの D12～D15(HSTC0～3)ビットは8段プリバッファのスタックカウンタ値に割り当てられていて、バッファの蓄積量を示しています。4ビットの値が0の時は完全な空状態を示しています。8になると満杯の状態を示していますので、それ以上セグメントデータを書き込むことはできません。補間ドライブ命令を書き込むとスタックカウンタ値は1つ増加し、現在出力中のセグメントのドライブが終了するとスタックカウンタ値は1つ減少します。
また、RR0 レジスタの D11(CNEXT)ビットは連続補間ドライブの次データの書込み可能を知らせるビットです。この CNEXT ビットは、補間ドライブが開始されると、プリバッファのスタックカウンタが1から7の間は1になります。上位 CPU はこのビットを確認し、1であれば次データを書き込んでも良いと判断できます。

(10) 第nセグメントのデータ・補間命令書き込み

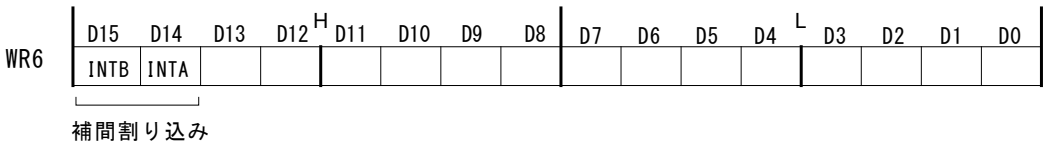
第9セグメント以降は補間ドライブ実行状態でセグメントデータを書き込んでいきます。書き込む内容は(4)および(5)に記されている第1～8セグメントと同様です。補間ドライブ命令を書き込んだら、(7)へ戻ります。

3.7.2 割り込みを用いた連続補間

連続補間は割り込みを用いても実現することができます。本ICのINT1N信号(端子番号:34)は、プリバッファに空きができるとLowアクティブになり、上位に対して次のセグメントデータの書き込みが可能であることを知らせます。空き状態を知らせる割り込みの発生タイミングは2種類設定する事ができます。

■ 補間割り込みの設定

プリバッファの空きを知らせる割り込みは、補間モード設定(2Ah)の D14,D15 の 2 ビットで設定します。



D14(INTA)ビットを1にすると、プリバッファのスタックカウンタが4から3に変化した時にINT1N信号がLowアクティブになります。8段プリバッファのおよそ半分が空になったことを知らせます。連続補間ドライブを比較的ゆっくり行う場合に向いています。

D15(INTB)ビットを1にすると、プリバッファのスタックカウンタが8から7に変化した時にINT1N信号がLowアクティブになります。プリバッファに1つ空きができたことを知らせます。連続補間ドライブを高速で行う場合に向いています。

■ 割り込み処理

INT1N信号による割り込みが発生したら、上位CPUは割り込み処理ルーチンにおいて、必要数だけの次セグメントデータを書き込みます。書き込む内容は上記(4)および(5)に記されている第1～8セグメントと同様です。1つのセグメントデータの最後には必ず補間ドライブ命令を書き込む必要があります。書き込みは、RR0レジスタのD15～12ビット(HSTC3～0)でスタックカウンタの値を確認しながら行うことも可能です。

■ 割り込み信号(INT1N)の解除

INT1N信号は、次の補間ドライブ命令を書き込むと自動的にクリアされ、hi-Zに戻ります。また、次の方法でもでもクリアされます。

- 補間割り込みクリア命令(6Fh)を発行
- 連続補間ドライブが終了する

3.7.3 連続補間時のエラー発生

連続補間時のエラー発生は、リミットオーバラン等のエラーと補間データ書き込みエラーの 2 種類があります。

■ リミットオーバーラン等のエラー

連続補間のドライブ途中でリミットオーバラン等のエラーが発生すると、現在ドライブ中の補間セグメントで停止します。エラーで停止するとプリバッファのスタックカウンタは0になりますので、すでに書き込まれているそれ以降のセグメントデータおよび補間命令はすべて無効になります。エラー解除後に続行させることはできません。

■ データ書き込みエラー

補間データ書き込みエラーは、現在補間ドライブ中のセグメントの次に続くセグメントのデータセットが間に合わなかったときに発生するエラーです。

連続補間では、プリバッファにセットされている最後のセグメントの補間ドライブ最終パルスの立下り(正論理時)以前に次に続くセグメントのデータセットおよび補間ドライブ命令の書込みが完了すれば問題ないのですが、最終パルスが立ち下がったあと、このセグメントドライブが終了する間に、次のセグメントの補間ドライブ命令書込みが行われても、データを処理できません。このとき、セットしようとしていたセグメントは実行されず、プリバッファのスタックカウンタもアップしません。主軸の RR2 レジスタの D7 ビット(補間エラー)に 1 が立ち、補間ドライブはエラーにより終了します。このエラーの解除は、補間軸全ての軸に対し、エラー終了・ステータスクリア命令(79h)を発行して行います。

3.7.4 連続補間の注意点

- 各補間セグメントは終点など必要なデータをセットしたのちに、補間ドライブ命令をセットします。順序を逆にすると正しく動作しません。
- 連続補間のドライブ速度は最高4MPPS(CLK=16MHz 時)までです。
- プリバッファにセットしているすべての補間セグメントをドライブする時間は、補間軸のエラーチェック、次の補間セグメントのデータおよび命令をセットする時間以上あることが必要です。もし、次の補間セグメントのデータをセットしている間に、現在の補間セグメントのドライブが終了した場合に次の補間セグメントのドライブ命令が書き込まれると、いったん停止後に、続けて連続補間が行われることになります。ただし、補間命令の書き込みエラー(補間エラー)が発生した場合は、連続補間は終了します。
- 連続補間では直線補間での全軸の終点が0や、円弧補間での両方の軸の中心点がともに0など、いずれの軸もドライブパルスが出力されないデータセットはできません。このようなデータがセットされると正常に補間動作を行なうことができなくなります。
- 連続補間のなかに円弧補間がある場合、円弧補間は終点の短軸値が真値より±1LSB ずれる場合がありますので、各セグメントの誤差が累積しないように、あらかじめ各々の円弧補間の終点を確認してから、連続補間を組み立ててください。
- 2軸補間から3軸補間など、軸数の異なる連続補間はできません。
- 連続補間の途中で、補間軸指定の変更はできません。
- エラーによる停止をした場合、必ずエラーの種別を確認し、その後、エラー終了・ステータスクリア命令(79h)を発行してエラー解除の動作を行ってください。エラー状態のままでは、補間ドライブを行う事はできません。
- 連続補間のドライブ中に停止命令で停止した場合、プリバッファにセットされているセグメントデータは全て無効になります。
- ビットパターン補間およびヘリカル補間については、他の補間ドライブと組み合わせた連続補間を行うことができません。
- 連続補間ドライブの途中でドライブ速度を変更することはできません。(連続補間ドライブの途中でドライブ速度を変更したい場合は、弊社技術部までお問い合わせください。)

3.7.5 連続補間の実例

図3.7-2は、(0,0)を始点として、セグメントS1から、セグメントS21までを連続補間する例です。2軸直線補間と円弧補間の組み合わせです。円弧補間は半径500と半径1000の1/4円です。補間速度は1000PPSの定速ドライブで、2軸高精度線速一定モードで行います。セグメントS1は(X0,Y6000)の位置からスタートすることとします。下表は各セグメントの補間命令と設定データ値です。

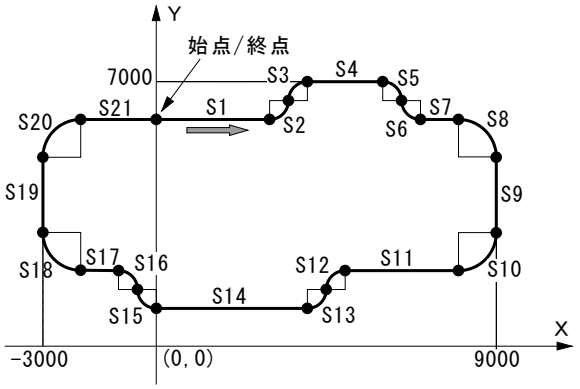


図 3.7-2 連続補間の例

セグメント番号	補間命令	終点X	終点Y	円弧中心X	円弧中心Y
S1	2軸直線	3000	0		
S2	CCW円弧	500	500	0	500
S3	CW円弧	500	500	500	0
S4	2軸直線	2000	0		
S5	CW円弧	500	-500	0	-500
S6	CCW円弧	500	-500	500	0
S7	2軸直線	1000	0		
S8	CW円弧	1000	-1000	0	-1000
S9	2軸直線	0	-2000		
S10	CW円弧	-1000	-1000	-1000	0
S11	2軸直線	-3000	0		
S12	CCW円弧	-500	-500	0	-500
S13	CW円弧	-500	-500	-500	0
S14	2軸直線	-4000	0		
S15	CW円弧	-500	500	0	500
S16	CCW円弧	-500	500	-500	0
S17	2軸直線	-1000	0		
S18	CW円弧	-1000	1000	0	1000
S19	2軸直線	0	2000		
S20	CW円弧	1000	1000	1000	0
S21	2軸直線	2000	0		

```
//---補間軸指定・モード設定 -----
      WR6 ← 00C3h ﾗｲﾄ       ; X,Y軸指定、２軸高精度線速一定
      WR0 ← 002Ah ﾗｲﾄ

//---補間ドライブ速度設定 -------
      WR6 ← 03E8h ﾗｲﾄ       ; 1000 PPS
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0104h ﾗｲﾄ       ; 主軸(X)に初速度を設定

      WR6 ← 03E8h ﾗｲﾄ       ; 1000 PPS
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0105h ﾗｲﾄ       ; 主軸にドライブ速度を設定

//---ドライブ開始ホールド -------
      WR0 ← 0177h ﾗｲﾄ       ; 主軸にドライブ開始ホールド

//--- セグメント１ ２軸直線  設定 -----
      WR6 ← 0BB8h ﾗｲﾄ       ; 終点Ｘ：3000
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0106h ﾗｲﾄ       ; 終点設定命令
      WR6 ← 0000h ﾗｲﾄ       ; 終点Ｙ：0
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0206h ﾗｲﾄ
      WR0 ← 0061h ﾗｲﾄ       ; ２軸直線補間命令

//--- セグメント２ CCW円弧  設定 -----
      WR6 ← 01F4h ﾗｲﾄ       ; 終点Ｘ：500
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0106h ﾗｲﾄ
      WR6 ← 01F4h ﾗｲﾄ       ; 終点Ｙ：500
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0206h ﾗｲﾄ
      WR6 ← 0000h ﾗｲﾄ       ; 円弧中心Ｘ：0
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0108h ﾗｲﾄ
      WR6 ← 01F4h ﾗｲﾄ       ; 円弧中心Ｙ：500
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0208h ﾗｲﾄ
      WR0 ← 0065h ﾗｲﾄ       ; CCW円弧補間命令

//--- セグメント３ CW円弧  設定 -----
      WR6 ← 01F4h ﾗｲﾄ       ; 終点Ｘ：500
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0106h ﾗｲﾄ
      WR6 ← 01F4h ﾗｲﾄ       ; 終点Ｙ：500
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0206h ﾗｲﾄ
      WR6 ← 01F4h ﾗｲﾄ       ; 円弧中心Ｘ：500
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0108h ﾗｲﾄ
      WR6 ← 0000h ﾗｲﾄ       ; 円弧中心Ｙ：0
```

```
      WR7 ← 0000h ﾗｲﾄ
      WR0 ← 0208h ﾗｲﾄ
      WR0 ← 0064h ﾗｲﾄ        ; CW円弧補間命令
```

セグメント4～8を同様にセットする。

```
//---ドライブ開始フリー -------
      WR0 ← 0178h ﾗｲﾄ        ; 主軸にドライブ開始フリー命令発行
                              ; 補間ドライブが開始される。
```

セグメントカウンタ SegCounter を９にセットする。

```
Loop:
//---エラーの確認 -------
      RR0 → ﾘｰﾄﾞ              ;
      RR0/D4またはD5が１ならばエラー発生。  エラー解析処理へ。

//---補間終了の確認 -------
      SegCounterが22ならば連続補間を終了する。

//---次セグメントデータ書き込み可を確認 -------
      RR0 → ﾘｰﾄﾞ              ;
      RR0/D11が１ならば書き込み可なので次へ。= 0ならば不可なので再度RR0をﾘｰﾄﾞ。

//---次セグメントデータ書き込み -------
     SegCounterの示すセグメントデータと補間命令を書き込む。

//---Loop へ戻る -------
     SegCounterを１つ増加させてLoopへジャンプする。
```

3.8 加減速ドライブでの補間

補間は、通常、定速ドライブで行いますが、本ICでは、直線加減速ドライブ、またはS字加減速ドライブ(直線補間のみ)で行うことも可能です。
補間ドライブでは、連続補間においても加減速ドライブを可能にするために、減速有効命令(6Dh)、減速無効命令(6Eh)を使用します。減速有効命令は、補間ドライブにおいて、自動減速、またはマニュアル減速を有効にする命令で、減速無効命令は、それを無効にする命令です。リセット時には無効になっています。加減速で単独の補間ドライブをするときには、ドライブ開始前に、必ず減速有効状態にしてください。ドライブの途中で減速有効命令を書き込んでも有効になりません。

3.8.1 直線補間の加減速ドライブ

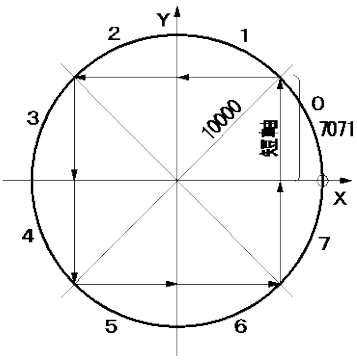
直線補間では、直線加減速ドライブおよびS字加減速ドライブが可能です。また、減速については、自動減速とマニュアル減速の両方が可能です。
マニュアル減速の場合は、終点座標の各軸の値のなかで絶対値が最も大きい値を主軸のマニュアル減速点として設定します。例えば、主軸:X, 第2軸:Y, 第3軸:Z軸において、終点(X:-20000,Y:30000,Z:-50000)までの3軸直線補間を行う場合、減速に必要とするパルス数を仮に5000とすると、Z軸の終点の絶対値が最も大きいので、50000-5000=45000を主軸X軸のマニュアル減速点にセットします。
直線補間の加減速ドライブの例は、3.1 節の3軸直線補間ドライブの例を参照してください。

【注意】短軸パルス均一化モードでは、S字加減速ドライブは使用できません。

3.8.2 円弧補間、ビットパターン補間の加減速ドライブ

円弧補間、ビットパターン補間では、マニュアル減速での直線加減速ドライブのみが可能です。S字加減速ドライブや、自動減速は使用できません。

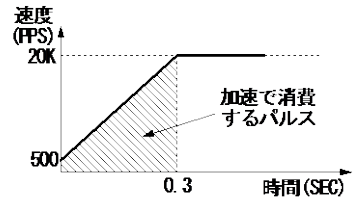
右図は、半径10000の真円の軌跡を直線加減速ドライブで描く例です。



円弧補間は自動減速できませんので、マニュアル減速点を、あらかじめ求める必要があります。

半径10000の円は、0から7象限すべてを通過します。各象限において、短軸となる軸は常にパルスを出力しますので、短軸側は1象限当たり10000／√2 =7071パルス出力することになります。従って、主軸から出力される基本パルスのパルス数は、円全体で、7071×8 = 56568となります。

また、初速度を500PPSとし、ドライブ速度20000PPSまでを0.3秒で直線加速させようとすると、加速度は (20000 - 500)/ 0.3 = 65000PPS/SECとなり、加速時に消費されるパルス数は 右下図の斜線部の面積になりますので、(500 +20000)× 0.3/ 2 = 3075となります。よって、減速度と加速度を同じとすれば、マニュアル減速点は 56568 - 3075 = 53493 に設定すれば良いことになります。



【注意】線速一定モードでは、上記の計算式は成り立ちません。

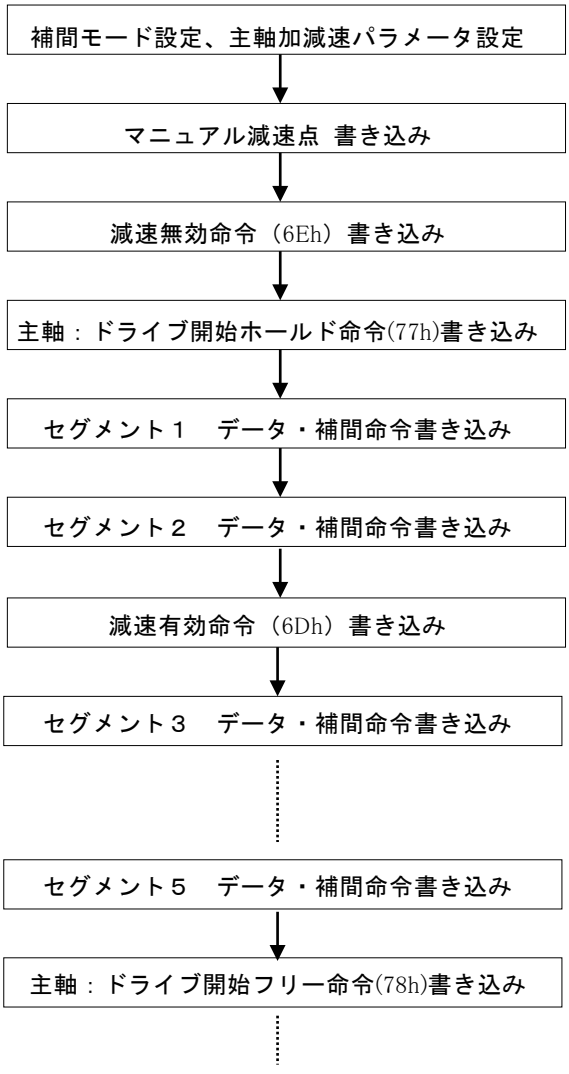
```
WR0 ← 011Fh ﾗｲﾄ        ; X軸選択
WR3 ← 0001h ﾗｲﾄ        ; 減速開始点：マニュアル

WR6 ← 0003h ﾗｲﾄ        ; 補間モード設定：X,Y軸指定
WR0 ← 002Ah ﾗｲﾄ

WR6 ← FDE8h ﾗｲﾄ        ; 加速度：65000 PPS/SEC
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0102h ﾗｲﾄ

WR6 ← 01F4h ﾗｲﾄ        ; 初速度：500 PPS
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 4E20h ﾗｲﾄ        ; ドライブ速度： 20000 PPS
WR0 ← 0105h ﾗｲﾄ
```

```
WR6 ← D8F0h ﾗｲﾄ       ; 中心Ｘ：-10000
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0108h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ       ; 中心Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0208h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ       ; 終点Ｘ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ

WR6 ← 0000h ﾗｲﾄ       ; 終点Ｙ：0
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ

WR6 ← D0F5h ﾗｲﾄ       ; マニュアル減速点：53493
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0107h ﾗｲﾄ

WR0 ← 006Dh ﾗｲﾄ       ; 減速有効

WR0 ← 0065h ﾗｲﾄ       ; ＣＣＷ円弧補間ドライブ
```

3.8.3 連続補間の加減速ドライブ

連続補間においても、マニュアル減速での直線加減速ドライブのみが可能です。S字加減速ドライブや、自動減速は使用できません。
連続補間では、あらかじめマニュアル減速点を設定しておかなければなりませんが、このマニュアル減速点は減速を開始するセグメントで出力される主軸からの基本パルスに対する値を設定します。
連続補間では、減速を開始するセグメントの補間データ設定時、補間命令を書き込む前に減速有効命令を書き込みます。減速を有効にしたセグメントのドライブに入ると減速有効状態になり、そのセグメントの開始からカウントしている主軸の基本パルスのパルス数がマニュアル減速点の値を越えたときに減速が開始されます。減速はセグメントにまたがって行うことができます。

例えば、セグメント1から5まである連続補間において、セグメント3からマニュアル減速させる場合には、次のような流れになります。



この例の場合、マニュアル減速点は、セグメント3開始からの主軸の基本パルスのパルス数に対する値ですので、ご注意ください。

3.9 補間ステップ送り

補間ドライブを、1パルスごとにステップ送りする動作です。コマンドで行う方法と外部信号で行う方法があります。外部信号を用いれば主軸からの基本パルスではなく、外部信号に同期した補間ドライブも可能です。

ステップ送りのときは、補間主軸は定速ドライブに設定します。各軸から出力されるドライブパルスのHiレベル幅は、補間の主軸で設定するドライブ速度によって決まるパルス周期の1／2の値になります。Lowレベル幅は次のコマンドまたは外部信号が来るまでのびることになります。図3.9-1は、外部信号による補間ステップ送りの例です。主軸の初速度を500PPS、ドライブ速度を500PPSの定速ドライブに設定すると、出力されるドライブパルスのHiレベル幅は1mSECになります。(ドライブパルスが正論理の場合)

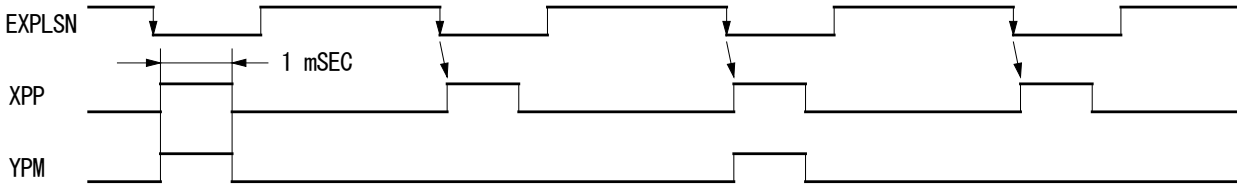


図 3.9-1 外部信号(EXPLSN)による補間ステップ送りの例(ドライブ速度:500PPS)

補間ステップ送りを使用する場合、補間モード設定命令(2Ah)で D9 ビットを1にすると、補間ステップ送りモードになります。

	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
WR6							STEP									

補間ドライブの外部信号・命令ステップ送り

3.9.1 コマンドによる補間ステップ送り

補間ドライブをステップ送りするコマンドとして、補間シングルステップ(6Fh)命令があります。
以下に操作手順を記述します。

a. 補間モード設定命令(2Ah)でD9ビットを1にする。
補間ステップ送りモードになります。

b. 補間の主軸の初速度とドライブ速度を同じ値で設定する。
初速度とドライブ速度を同じ値にすると定速ドライブになります。このときの速度値はシングルステップ命令を書き込むサイクルよりも速い速度に設定しなければなりません。例えば、シングルステップ命令を最高1mSECのサイクルで書き込む可能性があるならば、初速度とドライブ速度を1000PPSより速い値に設定します。

c. 補間のデータをセットする。(終点、中心点など)

d. 補間命令を書き込む。
補間命令を書き込んでもコマンドによる補間ステップモードになっていますので、各軸のドライブパルスは、まだ出力されません。

e. 補間ステップ(6Fh)命令を書き込む。
補間演算の結果のドライブパルスが各軸から出力されます。補間ドライブが終了するまで、補間ステップ(6Fh)命令を書き込みます。

補間ステップ送りを途中で中止する場合は、主軸に対して、即停止命令(57h)を書き込み、ドライブ速度での1パルス周期以上のタイムディレイをおいた後、再度、補間ステップ命令(6Fh)を書き込むと、ドライブが停止します。

補間ドライブ終了後に書き込まれた補間シングルステップ命令は、無効になります。

3.9.2 外部信号による補間ステップ送り

EXPLSN端子(30)は、補間ドライブをステップ送りするための外部入力信号です。
EXPLSN入力信号は、通常、Hiレベルにしておきます。外部信号による補間ステップモードでは、Lowレベルへの↓で補間ステップ送りが行われます。

以下に操作手順を記述します。

a. **補間モード設定(2Ah)でD9ビットを１にする。**
補間ステップモードになります。

b. **補間の主軸の初速度とドライブ速度を同じ値で設定する。**
初速度とドライブ速度を同じ値にすると定速ドライブになります。このときの速度値は、コマンドの場合と同様に、 EXPLSNのLowパルスのサイクルよりも速い速度に設定しなければなりません。

c. **補間のデータをセットする。（終点、中心点など）**

d. **補間命令を書き込む。**
補間命令を書き込んでも外部信号による補間ステップモードになっていますので、各軸のドライブパルスは、まだ出力されません。

e. EXPLSN**入力に**Low**レベルパルスを入力する。**
パルスの立ち下がりから2～5CLK後に、補間ドライブパルスが各軸から出力されます。

EXPLSNのLowレベルパルス幅は4CLK以上必要です。また、EXPLSNのパルス周期は、主軸に設定したドライブ速度の周期よりも、必ず長くなければなりません。

補間ドライブが終了するまで、EXPLSNのLowレベルパルスを繰り返します。

補間ステップ送りを途中で中止する場合は、主軸に対して、即停止命令(57h)を書き込み、ドライブ速度での1パルス周期以上のタイムディレイをおいた後、再度、EXPLSNのLowレベルパルスを入力すると、ドライブが停止します。(手っ取り早く、ソフトリセットをかけてしまう方法もあります。)

補間ドライブ終了後のEXPLSNのLowパルスの入力は、無効になります。

3.9.3 補間ステップ送りの注意点

- ESPLSN 信号はフィルタ機能がありません。EXPLSN の Low パルスをメカニカル接点で生成する場合は、チャッタリングによる誤動作が発生しないように対策してください。
- 補間ステップ送りでは、短軸パルス均一化は使用することができません。

3.10 マルチチップ補間

本ICを複数チップ使用して多軸の直線補間を行う機能です。
図3.10-1は、3チップ使用して12軸の直線補間を行う場合の接続例です。メインチップは、サブチップに対して補間ドライブの同期パルスを送出する役割を持っていますので、補間ドライブ速度のパラメータはメインチップの主軸に設定します。
図に示すように、8本のマルチチップ補間用信号(MPLS,MCLK,MERR,MINP,MDT3～0)をチップ間で各々接続して、抵抗(3.3KΩ程度)でプルアップします。これらの信号は汎用入力信号(PIN7～0)と兼用になっていますので、汎用入力信号としては、使用できなくなります。

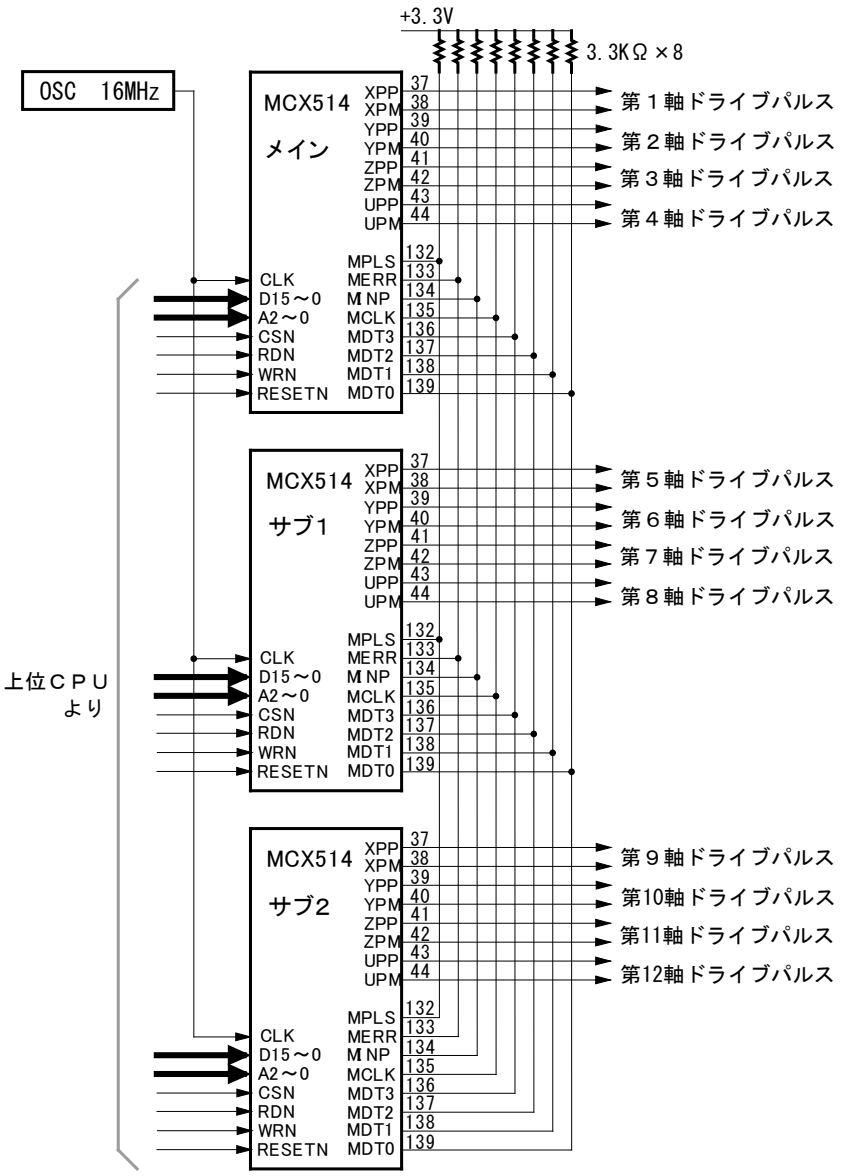


図 3.10-1 マルチチップ多軸補間の接続例

各信号は次のような働きをします。

表 3.10-1 マルチチップ補間の各信号の働き

信号名(端子番号)	信号機能	通信方向	兼用汎用入力信号
MPLS(132)	補間ドライブの同期パルス	メイン→サブ	PIN7
MERR(133)	エラー発生／メインチップの停止	メイン←→サブ	PIN6
MINP(134)	インポジション待ち	メイン←サブ	PIN5
MCLK(135)	MDT3～0 のデータ転送クロック	メイン←→サブ	PIN4
MDT3～0(136～139)	各チップの終点の転送データ	メイン←→サブ	PIN3～0

3.10.1 実行手順

以下、複数チップによる多軸直線補間の手順を記述します。

(1) マルチチップメイン・サブの指定および補間軸指定

補間モード設定命令(2Ah)によってメインチップ・サブチップの指定、およびそれぞれのチップ内の補間実行軸の指定を行います。補間モード設定命令は WR6 レジスタの所定ビットを設定してから、WR0 レジスタに命令コード 2Ah を書き込むと実行されます。WR6 レジスタの他の補間モード指定ビットも必要に応じて設定します。

WR6	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	H												L			
	INTB	INTA	0	MAXM	MLT1	MLT0	STEP	LMDF	SPD1	SPD0	0	CXIV	U-EN	Z-EN	Y-EN	X-EN
					メイン・サブ指定								補間軸指定			

メインチップ・サブチップの指定は、D10,11 ビット(MLT0,1)で設定します。

表 3.10-2 マルチチップ補間のチップ指定

D11(MLT1)	D10(MLT0)	メイン・サブチップ指定
0	0	マルチチップ補間を行わない。
0	1	メインチップとしてマルチチップ補間を行う。
1	0	サブチップとしてマルチチップ補間を行う。
1	1	無効（設定不可）

各チップ内の補間を行う軸は WR6/D3～D0 ビットで指定します。対応するビットを 1 にすると補間軸として有効になります。

表 3.10-3 マルチチップ補間の軸指定

WR6 のビット	補間軸
D0(X-EN)	X軸
D1(Y-EN)	Y軸
D2(Z-EN)	Z軸
D3(U-EN)	U軸

0：補間無効
1：補間有効

マルチチップ補間では、1 軸だけの指定も可能です。

【注意】マルチチップ補間では、短軸パルス均一化は使用することができません。D8 ビットは必ず 0 を指定してください。

(2) メインチップの主軸に対してドライブ速度パラメータ設定

補間ドライブの速度パラメータ設定をメインチップの主軸に対して行います。メインチップの他の補間軸やサブチップの補間軸には、設定をする必要がありません。メインチップの主軸には、加減速動作によって下表に示す必要な速度パラメータを設定します。

表 3.10-4 メインチップの主軸に設定する速度パラメータ

○が設定する必要あり

加減速動作	メインチップの主軸に設定する速度パラメータ				
	加速度増加率	加速度	減速度	初速度	ドライブ速度
定速ドライブ				○	○
台形加減速ドライブ		○		○	○
非対称台形加減速ドライブ		○	○	○	○
S字加減速ドライブ	○	○		○	○

【注意】
- マルチチップ補間のドライブ速度は、最高 4MPPS までです。ドライブ速度は 4MPPS 以下の値を設定してください。
- マルチチップ多軸直線補間のすべての軸の中で、最も移動パルス数の大きい軸が設定されたドライブ速度になります。メインチップの主軸が設定されたドライブ速度になるわけではありません。
- 台形およびS字加減速ドライブを行う場合は、必ず、補間ドライブ命令発行前に、メインチップに対して減速有効命令(6Dh)を発行しておく必要があります。

(3) 各軸の終点の設定

メイン・サブすべてのチップの補間を行うすべての各軸に対して、終点データを、現在値からの相対値で書き込みます。
マルチチップ補間の終点データ範囲は符号付28ビットです。WR6,7レジスタに終点データを書き込み、WR0レジスタに軸指定とともに命令コード06hを書き込むと設定されます。
一般に、多軸の直線補間を行う場合には、すべての軸の終点データの中の最大値が、各軸の直線補間演算において必要となります。本ICは、高速の連続直線補間を可能にするために、各軸の終点データをセットすると自動的に最大値が生成されるようにしています。CPUでの最大値の算出と各軸への最大値の設定は必要ありません。
あるチップの軸に終点データが書き込まれると、その書き込まれたチップから、マルチチップ補間用信号(MCLK,MDT3～0)を通して、その他のチップに終点データが転送されます。受信側チップでは、終点データを受け取ると、自チップ内の終点最大値の値と絶対値比較して、その値より大きければ終点最大値の値を更新します。

終点データの転送時間は、約2μsec(CLK=16MHz時)必要とします。従って、各軸の終点データの書き込みの間隔はこの時間より短くできません。高速演算CPUで、終点データの書き込みサイクルがこの時間より速い場合にはソフト的に遅延を入れる必要があります。

終点最大値は、リセット時および補間ドライブ命令開始直後に値が0にクリアされます。また、終点最大値クリア命令(7Ch)でもクリアできます。また、終点最大値は、終点最大値読み出し命令(39h)で読み出すことができますので、全軸の終点データ書き込み後に、正しく最大値が生成されているか確認することができます。

【注意】終点最大値読み出し命令(39h)で読み出す値は、補間ドライブ実行前と実行中で内容が異なります。終点最大値読み出し命令(39h)の詳細は、7.4.10 項を参照してください。

■ 終点データ転送エラー

終点データ受信側の各チップにおいては、データサムおよび転送フレームに異常がないかをチェックしています。受信が正常に行われなかった場合はエラーとなり、RR2レジスタのD7ビット(CERR)およびRR3レジスタ ページ1のD12ビット(MCERR)が1になります。また、RR0レジスタの補間軸全てのエラービット(D7～4:n-ERRの該当ビット)が1になります。サブチップで受信エラーが発生した場合、マルチチップ補間用信号(MERR)を経由してエラー発生をメインチップに伝え、メインチップのRR0レジスタの主軸のエラービットも1になります。

(4) 補間命令の書き込み

それぞれのサブチップに対して、そのチップの補間軸数に応じた直線補間命令(60h～63h)を発行します。続いて、メインチップに対して補間軸数に応じた直線補間命令(60h～63h)を発行します。メインチップをサブより先に発行すると正常に動作しません。
加減速ドライブを行う場合は、必ず、補間ドライブ命令発行前に、メインチップに対して減速有効命令(6Dh)を発行しておく必要があります。

表 3.10-5 マルチチップ補間命令

補間命令	命令コード
1 軸直線補間ドライブ	60h
2 軸直線補間ドライブ	61h
3 軸直線補間ドライブ	62h
4 軸直線補間ドライブ	63h

メインチップに直線補間命令(60h～63h)が発行されると、メインチップは、直ちにMPLS信号から各サブチップに対して補間ドライブの同期パルスの送出を開始し、全軸の直線補間が開始されます。

(5) ドライブ終了、エラー確認

補間ドライブ中は各チップのRR0レジスタにおいて補間有効軸のドライブビット(D3～0:n-DRV)が1になります。補間ドライブの終了はメインチップの主軸のドライブビットが0に戻ることで判断します。
各軸のインポジションを有効に設定した場合には、有効にしたすべての軸のnINPOS信号がアクティブレベルになるのを待ってからメインチップRR0レジスタの主軸のドライブビット(D3～0:n-DRVの主軸のビット)が0に戻ります。

補間ドライブ中に、メインチップの各軸にエラーが発生すると、各軸のRR2レジスタのD5～0ビットのどれかが1になるとともに、RR0レジスタのエラービット(D7～4:n-ERRで該当する軸)が1になります。
また、サブチップの各軸においてエラーが発生すると、メインチップ同様に各軸のRR2レジスタのD5～0ビットのどれかが1になるとともに、RR0レジスタのエラービット(D7～4:n-ERRで該当する軸)が1になります。さらに、サブチップはマルチチップ補間用信号のMERR信号をLowアクティブにしてメインチップにエラー発生を伝えます。メインチップではエラーを受信すると、RR0レジスタの主軸のエラービット(D7～4:n-ERRで該当する軸)が1になります。メインチップは、エラーが発生するとサブチップに対して補間ドライブ同期パルスの送出を停止しますので、全軸が即停止します。
従って、補間ドライブ中および補間終了時のエラーの確認は、メインチップ主軸のRR0レジスタエラービット(D7～4:n-ERR)だけを監視してしていれば良いことになります。エラーが検出(ビットデータ =1)されたら、補間を実施している各軸のRR2レジスタ(エラー表示レジスタ)の内容を確認しエラー原因を解析する処理を行ってください。

3.10.2 補間ドライブの途中停止

補間ドライブを途中で停止させたい時には、メインチップの主軸に対してドライブ停止命令を発行してやります。主軸のドライブが停止すると、メインチップの他の補間実施軸およびサブチップの補間実施軸もドライブが停止し、RR0レジスタのドライブビット(D3～0:n-DRVの該当ビット)が0に戻ります。

3.10.3 連続補間

マルチチップ補間も直線補間を連続して行うことができます。連続補間の実行方法は、単一チップの連続補間と同様です。ドライブ開始ホールド、ドライブ開始フリー命令は、メインチップの主軸に書き込みます。メインチップのプリバッファに空きがあれば、すべての補間軸に終点データを書き込むことができます。サブチッププリバッファの空き状態もメインチップと同じタイミングで増減します。

3.10.4 マルチチップ補間の注意点

- マルチチップ補間用信号(MPLS,MCLK,MERR,MINP,MDT3～0)は、必ず抵抗で 3.3V 電源にプルアップしてください。抵抗値の範囲は 1K～5.1KΩですが、3.3KΩ程度を推奨します。
- マルチチップ補間用信号(MPLS,MCLK,MERR,MINP,MDT3～0)の配線は、できるだけ短く配線して、他の信号と交錯しないようにしてください。お客様の回路システムにおいて、ジャンパー切り替えなどにより汎用入力信号と兼ねることは避けてください。
- マルチチップ補間では、メインチップの軸のみ線速一定にすることができます。
- 連続補間ではインポジションは無効のままにしておいてください。

3.10.5 マルチチップ補間の実例

メインチップとサブチップ2個の実例を示します。

■ 例 1 各チップ X 軸,Y 軸の 2 軸によるマルチチップ補間

【プログラム例】

```
// メインチップ、サブチップの補間モードの設定
// メインチップへの書き込み
WR6 ← 0403h ﾗｲﾄ                    // メインチップ  X,Y 補間軸指定
WR0 ← 002Ah ﾗｲﾄ
// サブチップ 1 への書き込み
WR6 ← 0803h ﾗｲﾄ                    // サブチップ  X,Y 補間軸指定
WR0 ← 002Ah ﾗｲﾄ
// サブチップ 2 への書き込み
WR6 ← 0803h ﾗｲﾄ                    // サブチップ  X,Y 補間軸指定
WR0 ← 002Ah ﾗｲﾄ

// メインチップ主軸に対し、ドライブ関連パラメータ設定(2M PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                    // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 8480h ﾗｲﾄ                    // ドライブ速度  2M PPS
WR7 ← 001Eh ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

// 終点データの書き込みと受信エラーチェック
// メインチップへの書き込み
WR6 ← 0014h ﾗｲﾄ                    // 終点 1 X  20
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
// サブチップ 1 の受信エラーチェック                    Ａ処理とする
RR0 ／ D4,D5 ﾘｰﾄﾞ
D4,D5=1 なら ERROR(ｻﾌﾞﾁｯﾌﾟ)へｼﾞｬﾝﾌﾟ// エラー処理へ
// サブチップ 2 の受信エラーチェック                    Ｂ処理とする
RR0 ／ D4,D5 ﾘｰﾄﾞ
D4,D5=1 なら ERROR(ｻﾌﾞﾁｯﾌﾟ)へｼﾞｬﾝﾌﾟ// エラー処理へ
WR6 ← 000Ah ﾗｲﾄ                    // 終点 1 Y  10
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ａ処理の実行
//  Ｂ処理の実行

// サブチップ 1 への書き込み
WR6 ← FFF6h ﾗｲﾄ                    // 終点 1 X  -10
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
// メインチップの受信エラーチェック                    Ｃ処理とする
RR0 ／ D4,D5 ﾘｰﾄﾞ
D4,D5=1 なら ERROR(ﾒｲﾝﾁｯﾌﾟ)へｼﾞｬﾝﾌﾟ// エラー処理へ
//  Ｂ処理の実行
WR6 ← 0005h ﾗｲﾄ                    // 終点 1 Y  5
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ｃ処理の実行
//  Ｂ処理の実行

// サブチップ 2 への書き込み
WR6 ← 0019h ﾗｲﾄ                    // 終点 1 X  25
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ｃ処理の実行
//  Ａ処理の実行
WR6 ← FFF4h ﾗｲﾄ                    // 終点 1 Y  -12
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ｃ処理の実行
//  Ａ処理の実行

// サブチップ、メインチップの順に補間命令を発行
// サブチップ 1 への書き込み
WR0 ← 0061h ﾗｲﾄ                    // 2 軸直線補間
// サブチップ 2 への書き込み
WR0 ← 0061h ﾗｲﾄ                    // 2 軸直線補間
// メインチップへの書き込み
WR0 ← 0061h ﾗｲﾄ                    // 2 軸直線補間
```

```
// ERROR 処理(メインチップ)
WR0 ← 011Fh ﾗｲﾄ                    // 補間軸のいずれかに対し(例では X 軸)、RR2 レジスタの補間エラー確認
RR0 ／ D4,5  ﾘｰﾄﾞ                   // 補間軸のエラー確認
RR2 ／ D7    ﾘｰﾄﾞ                   // 補間エラー確認
WR0 ← 017Bh ﾗｲﾄ                    // 補間軸のいずれかに対し(例では X 軸)、RR3 のページ 1 表示命令発行
RR3 ／ D12   ﾘｰﾄﾞ                   // マルチチップ補間転送エラー確認

WR0 ← 0179h ﾗｲﾄ                    // 補間軸に対してエラークリア命令発行
RR0 ／ D4,5  ﾘｰﾄﾞ                   // 補間軸のエラークリア確認
RR2 ／ D7    ﾘｰﾄﾞ                   // 補間エラークリア確認
RR3 ／ D12   ﾘｰﾄﾞ                   // マルチチップ補間転送エラークリア確認

// ERROR 処理(サブチップ)
WR0 ← 011Fh ﾗｲﾄ                    // 補間軸のいずれかに対し(例では X 軸)、RR2 レジスタの補間エラー確認
RR0 ／ D4,5  ﾘｰﾄﾞ                   // 補間軸のエラー確認
RR2 ／ D7    ﾘｰﾄﾞ                   // 補間エラー確認
WR0 ← 017Bh ﾗｲﾄ                    // 補間軸のいずれかに対し(例では X 軸)、RR3 のページ 1 表示命令発行
RR3 ／ D12   ﾘｰﾄﾞ                   // マルチチップ補間転送エラー確認
// メインチップの RR0 レジスタの読み出し
RR0 ／ D4    ﾘｰﾄﾞ                   //  メインチップの主軸のエラー確認

WR0 ← 0179h ﾗｲﾄ                    // 補間軸に対してエラークリア命令発行
RR0 ／ D4,5  ﾘｰﾄﾞ                   // 補間軸のエラークリア確認
RR2 ／ D7    ﾘｰﾄﾞ                   // 補間エラークリア確認
RR3 ／ D12   ﾘｰﾄﾞ                   // マルチチップ補間転送エラークリア確認
// メインチップの RR0 レジスタの読み出し
RR0 ／ D4    ﾘｰﾄﾞ                   //  メインチップの主軸のエラークリア確認
```

■ 例 2 マルチチップ補間の連続補間

下の例のA、B、C処理およびERROR処理は、例1と同じ処理を行います。

【プログラム例】

```
// メインチップ、サブチップの補間モードの設定
// メインチップへの書き込み
WR6 ← 0403h ﾗｲﾄ                    // メインチップ  X,Y 補間軸指定
WR0 ← 002Ah ﾗｲﾄ
// サブチップ 1 への書き込み
WR6 ← 0803h ﾗｲﾄ                    // サブチップ  X,Y 補間軸指定
WR0 ← 002Ah ﾗｲﾄ
// サブチップ 2 への書き込み
WR6 ← 0803h ﾗｲﾄ                    // サブチップ  X,Y 補間軸指定
WR0 ← 002Ah ﾗｲﾄ

// メインチップ主軸に対し、ドライブ関連パラメータ設定(2M PPS の定速ドライブの設定)
WR6 ← 1200h ﾗｲﾄ                    // 初速度  8M PPS (仕様最大)
WR7 ← 007Ah ﾗｲﾄ
WR0 ← 0104h ﾗｲﾄ

WR6 ← 8480h ﾗｲﾄ                    // ドライブ速度  2M PPS
WR7 ← 001Eh ﾗｲﾄ
WR0 ← 0105h ﾗｲﾄ

// メインチップ主軸に対し、ドライブ開始ホールド命令発行
WR0 ← 0177h ﾗｲﾄ

// 終点データの書き込みと受信エラーチェック
// Seg1
// メインチップへの書き込み
WR6 ← 0014h ﾗｲﾄ                    // 終点 1 X  20
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ａ処理の実行
//  Ｂ処理の実行
WR6 ← 000Ah ﾗｲﾄ                    // 終点 1 Y  10
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ａ処理の実行
//  Ｂ処理の実行

// サブチップ 1 への書き込み
WR6 ← FFF6h ﾗｲﾄ                    // 終点 1 X  -10
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ｃ処理の実行
//  Ｂ処理の実行
WR6 ← 0005h ﾗｲﾄ                    // 終点 1 Y  5
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ｃ処理の実行
//  Ｂ処理の実行

// サブチップ 2 への書き込み
WR6 ← 0019h ﾗｲﾄ                    // 終点 1 X  25
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ｃ処理の実行
//  Ａ処理の実行
WR6 ← FFF4h ﾗｲﾄ                    // 終点 1 Y  -12
WR7 ← FFFFh ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ｃ処理の実行
//  Ａ処理の実行

// サブチップ、メインチップの順に補間命令を発行
// サブチップ 1 への書き込み
WR0 ← 0061h ﾗｲﾄ                    // 2 軸直線補間
// サブチップ 2 への書き込み
WR0 ← 0061h ﾗｲﾄ                    // 2 軸直線補間
// メインチップへの書き込み
WR0 ← 0061h ﾗｲﾄ                    // 2 軸直線補間
```

```
// Seg2
// メインチップへの書き込み
WR6 ← 000Ah ﾗｲﾄ                    // 終点１X  10
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ａ処理の実行
//  Ｂ処理の実行
WR6 ← 0014h ﾗｲﾄ                    // 終点１Y  20
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ａ処理の実行
//  Ｂ処理の実行

// サブチップ１への書き込み
WR6 ← 0005h ﾗｲﾄ                    // 終点１X  5
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ｃ処理の実行
//  Ｂ処理の実行
WR6 ← 000Ah ﾗｲﾄ                    // 終点１Y  10
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ｃ処理の実行
//  Ｂ処理の実行

// サブチップ２への書き込み
WR6 ← 0019h ﾗｲﾄ                    // 終点１X  25
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0106h ﾗｲﾄ
//  Ｃ処理の実行
//  Ａ処理の実行
WR6 ← 000Ch ﾗｲﾄ                    // 終点１Y  12
WR7 ← 0000h ﾗｲﾄ
WR0 ← 0206h ﾗｲﾄ
//  Ｃ処理の実行
//  Ａ処理の実行

// サブチップ、メインチップの順に補間命令を発行
// サブチップ１への書き込み
WR0 ← 0061h ﾗｲﾄ                    // ２軸直線補間
// サブチップ２への書き込み
WR0 ← 0061h ﾗｲﾄ                    // ２軸直線補間
// メインチップへの書き込み
WR0 ← 0061h ﾗｲﾄ                    // ２軸直線補間

// 以下、最大 Seg8 まで必要な分、繰り返す
・
・
・

// メインチップ主軸に対し、ドライブ開始フリー命令発行
WR0 ← 0178h ﾗｲﾄ                    // 連続補間ドライブ開始
```

4. I²C シリアルバス

本ICは上位CPUとの接続インターフェイスとして、従来の8ビット/16ビットデータバスの他に、I²C シリアルバスを備えています。I²C シリアルバスは、2本のみの信号線(シリアル・データライン(SDA) とシリアル・クロックライン(SCL))で情報の転送を行います。データ転送速度は、バスの負荷容量 400pF 以下において、標準モード(100Kbit/sec)、ファーストモード(400Kbit/sec)、ファーストプラスモード(1Mbit/sec)まで可能です。データ転送効率を 8 ビット/16 ビットデータバスと比較すると、数10倍から100倍程度遅くなりますが、必要なパラメータ(ドライブ速度、移動パルス数など)を設定して相対移動ドライブを開始するまでを1～数msecの時間以内には行うことができます。それほど高速のセットアップを要求しないシステムには、非常に都合の良いバスインターフェイスと言えます。
下図は、I²C シリアルバスの接続例を示しています。

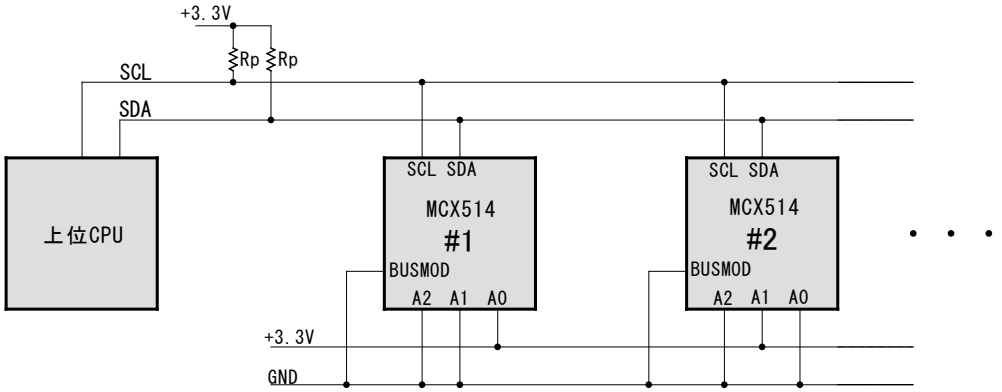


図 4.1-1 I²C シリアルバス接続例

4.1 I²C バスモードで使用する端子

本 IC を I²C バスモードで使用するためには、下表の端子を適切に接続する必要があります。

表 4.1-1 I²C バスモード時の接続端子

信号名	端子番号	説 明
BUSMOD	32	本 IC のバスモードを設定します。Low レベルにすると I²C バスモードになります。
A2～A0	22～24	アドレス信号の A2～A0(22～24)をチップアドレス設定端子として使用します。Low レベルが０、Hi レベルが１に設定されます。 同一バス上に接続できる MCX514 チップは最大８個になります。
SDA	25	I²C バスの SDA 信号端子です。プルアップ抵抗が必要です。
SCL	26	I²C バスの SCL 信号端子です。プルアップ抵抗が必要です。 CSN 信号との兼用端子です。I²C バスモードとすると SCL 信号入力になります。
I2CRSTN	31	本 IC の I²C 制御部をリセットする信号です。CLK 非同期入力で Low レベルにするとリセットされます。１μsec 以上 Low にしてください。 H16L8 信号との兼用端子です。

4.1.1 プルアップ抵抗(Rp)

バスラインの SDA および SCL 信号にはプルアップ抵抗(Rp)が必要です。プルアップ抵抗の値は、データ転送速度とバスの負荷容量に依存します。詳細については NXP 社の I²C バス規格書を参照してください。

4.1.2 I2CRSTN リセット

■ 初期設定時

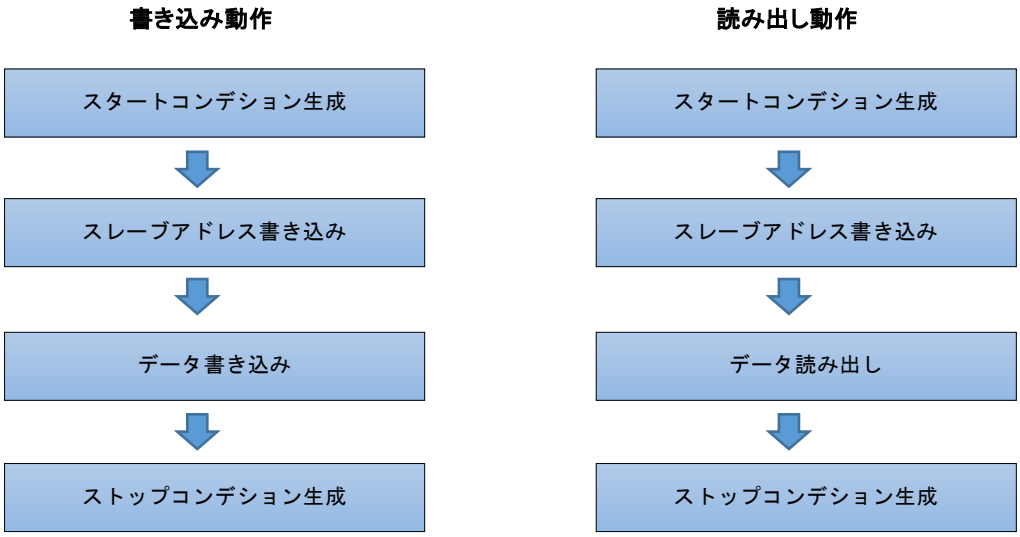
システムの初期状態において、上位 CPU 側の I²C 端子モードの切り替えなどによって SCL 信号および SDA 信号にノイズが発生し、その後のデータ転送が正常に行われないことがあります。CPU 側の I²C 初期設定手順を変えるなどして、ノイズの改善を試みてください。それでもノイズの改善が見られない時には、I²C 初期設定完了後に MCX514 に対して I2CRSTN 信号による I²C リセットをかける必要があります。また、RESETN 信号による MCX514 の本体リセットでも I²C 制御部はリセットされますので、RESETN 信号で代用することも可能です。

■ データ転送時

アクノリッジ信号が Hi を返すなど、I²C 通信が正常に動作しなかった場合、I2CRSTN 信号で I²C 制御部のリセットを行ってください。上記同様、RESETN 信号で代用することも可能です。

4.2 I²C バス送受信手順

上位 CPU から MCX514 の WR レジスタへの書き込みおよび RR レジスタからの読み出し手順は次のようになります。



MCX514 はスレーブの機能しか持ちません。

4.2.1 書き込み動作

MCX514 の WR レジスタへの書き込み動作の手順について説明します。

■ スタートコンデション生成

SCL 信号が Hi のときに SDA 信号が Hi から Low に変化するとスタートコンデションとなります。いかなる送受信においても CPU 側が初めにこのコンデションを作る必要があります。

■ スレーブアドレス書き込み

スタートコンデションの次に、どのチップのどの WR レジスタから書き込みを行うかの指定を MCX514 に送信します。下記に示す SCL に同期した 8 ビットのスレーブアドレスを送信し、9 ビット目に MCX514 からの ACK(Low)を受け取ります。スレーブアドレスの構成は、D7～D5 の 3 ビットはチップアドレス、D4～D1 の 4 ビットはレジスタアドレス、最後の D0 は読み出し/書き込み指定ビットから成ります。

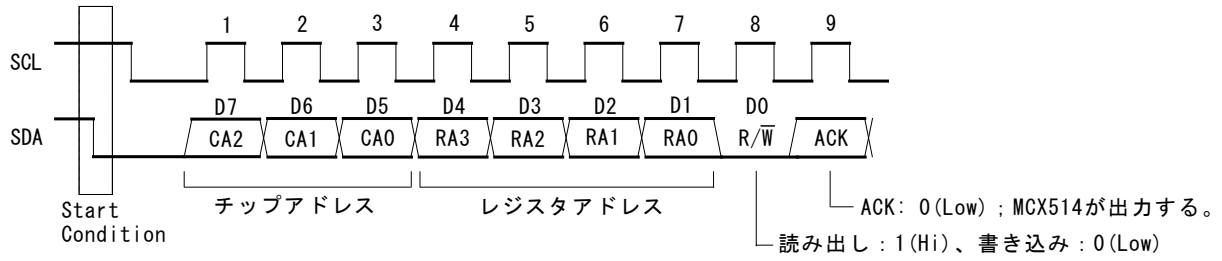


図 4.2-1 スレーブアドレス

チップアドレス CA2～CA0 は、MCX514 の A2(22)、A1(23)、A0(24)端子で設定したアドレスを指定します。端子設定が Low は0に、Hi は1に対応します。
レジスタアドレスは書き込みを開始したいレジスタアドレスを下表を参照して指定します。WR レジスタは 16 ビット構成ですが、I^2C の送信データではバイト単位の指定になります。

表 4.2-1 書き込みレジスタアドレス

レジスタアドレス				WRn レジスタ
RA3	RA2	RA1	RA0	
0	0	0	0	WR0L
0	0	0	1	WR0H
0	0	1	0	WR1L
0	0	1	1	WR1H
0	1	0	0	WR2L
0	1	0	1	WR2H
0	1	1	0	WR3L
0	1	1	1	WR3H
1	0	0	0	WR4L
1	0	0	1	WR4H
1	0	1	0	WR5L
1	0	1	1	WR5H
1	1	0	0	WR6L
1	1	0	1	WR6H
1	1	1	0	WR7L
1	1	1	1	WR7H

WRnL は WRn の下位バイト(D7～D0)です。
WRnH は WRn の上位バイト(D15～D8)です。

スレーブアドレス書き込みの最後のビット D0 は読み出し/書き込みの指定ビットです。書き込みを行う場合には0に設定します。

スレーブアドレスを 8SCL で送信すると、第9SCL 目に MCX514 側から ACK 情報を SDA 信号に返します。正しく 8 ビットのスレーブアドレスを受け取り、チップアドレスが一致した MCX514 は、Low(オープンドレイン出力を ON)を返します。正しく受け取れなかったり、チップアドレスが不一致の場合には Low を返しません。

■ データ書き込み

スレーブアドレスの書き込みに続いて、データの書き込みを行います。書き込みデータはスレーブアドレスで指定した WRn レジスタから 1 バイト単位で送信します。1 バイトだけの書き込みから、必要なバイト数を複数連続して書き込むことが可能です。1 バイト送信の後に第9SCL 目で、MCX514 は正しく受信できると Low レベルの ACK 信号を SDA ライン上に返します。CPU 側はこの ACK 信号を受け取った後に次のレジスタアドレスに書き込む 1 バイトデータを送信します。

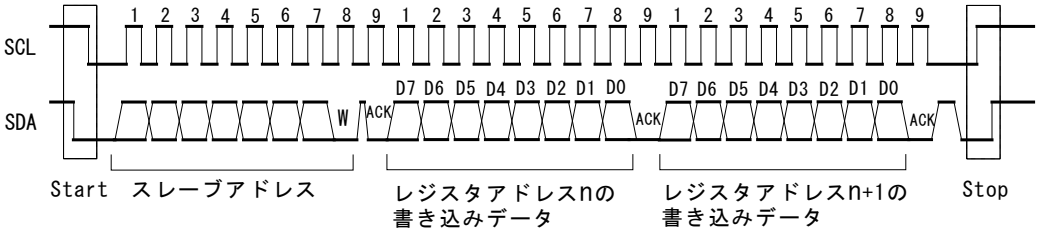


図 4.2-2 データ書き込み

■ ストップコンデション生成

データ書き込みを終了させる時にはストップコンデションを生成します。SCL 信号が Hi のときに SDA 信号が Low から Hi に変化するとストップコンデションとなります。いかなる送受信においても CPU 側が最後にこのコンデションを作る必要があります。

4.2.2 読み出し動作

MCX514 の RR レジスタからの読み出し動作の手順について説明します。

■ スタートコンデション生成

SCL 信号が Hi のときに SDA 信号が Hi から Low に変化するとスタートコンデションとなります。いかなる送受信においても CPU 側が初めにこのコンデションを作る必要があります。

■ スレーブアドレス書き込み

スタートコンデションの次に、どのチップのどの RR レジスタから読み出しを行うかの指定を MCX514 に送信します。下記に示す SCL に同期した 8 ビットのスレーブアドレスを送信し、9 ビット目に MCX514 からの ACK(Low)を受け取ります。スレーブアドレスの構成は、D7～D5 の 3 ビットはチップアドレス、D4～D1 の 4 ビットはレジスタアドレス、最後の D0 は読み出し/書き込み指定ビットから成ります。

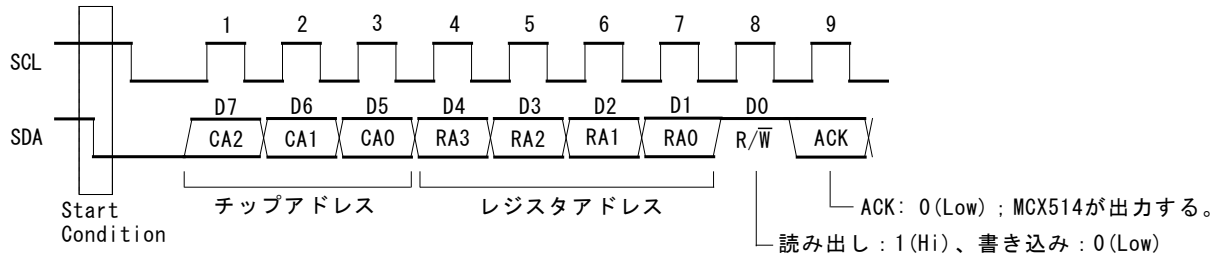


図 4.2-3 スレーブアドレス

チップアドレス CA2～CA0 は、MCX514 の A2(22)、A1(23)、A0(24)端子で設定したアドレスを指定します。端子設定が Low は0に、Hi は1に対応します。
レジスタアドレスは読み出しを開始したいレジスタアドレスを下表を参照して指定します。RR レジスタは 16 ビット構成ですが、I^2C の送信データではバイト単位の指定になります。

表 4.2-2 読み出しレジスタアドレス

レジスタアドレス				RRn レジスタ
RA3	RA2	RA1	RA0	
0	0	0	0	RR0L
0	0	0	1	RR0H
0	0	1	0	RR1L
0	0	1	1	RR1H
0	1	0	0	RR2L
0	1	0	1	RR2H
0	1	1	0	RR3L
0	1	1	1	RR3H
1	0	0	0	RR4L
1	0	0	1	RR4H
1	0	1	0	RR5L
1	0	1	1	RR5H
1	1	0	0	RR6L
1	1	0	1	RR6H
1	1	1	0	RR7L
1	1	1	1	RR7H

RRnLは RRn の下位バイト(D7～D0)です。
RRnH は RRn の上位バイト(D15～D8)です。

スレーブアドレス書き込みの最後のビット D0 は読み出し/書き込みの指定ビットです。読み出しを行う場合には1に設定します。

スレーブアドレスを 8SCL で送信すると、第9SCL 目に MCX514 側から ACK 情報を SDA 信号に返します。正しく 8 ビットのスレーブアドレスを受け取り、チップアドレスが一致した MCX514 は、Low(オープンドレイン出力を ON)を返します。正しく受け取れなかったり、チップアドレスが不一致の場合には Low を返しません。

■ データ読み出し
スレーブアドレスの書き込みに続いて、データの読み出しを行います。読み出しデータはスレーブアドレスで指定した RRn レジスタから 1 バイト単位で MCX514 が SDA ライン上に出力します。1 バイトだけの読み出しから、必要なバイト数を複数連続して読み出すことが可能です。1 バイト受信の後に第9SCL 目で、CPU 側は正しく受信できた場合 Low レベルの ACK 信号を SDA ライン上に返す必要があります。ただし、次にストップコンデションがくる最終データでは、Hi レベルの ACK を返してください。

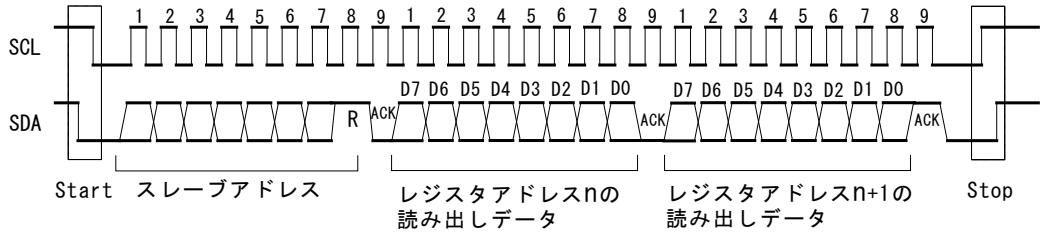


図 4.2-4 データ読み出し

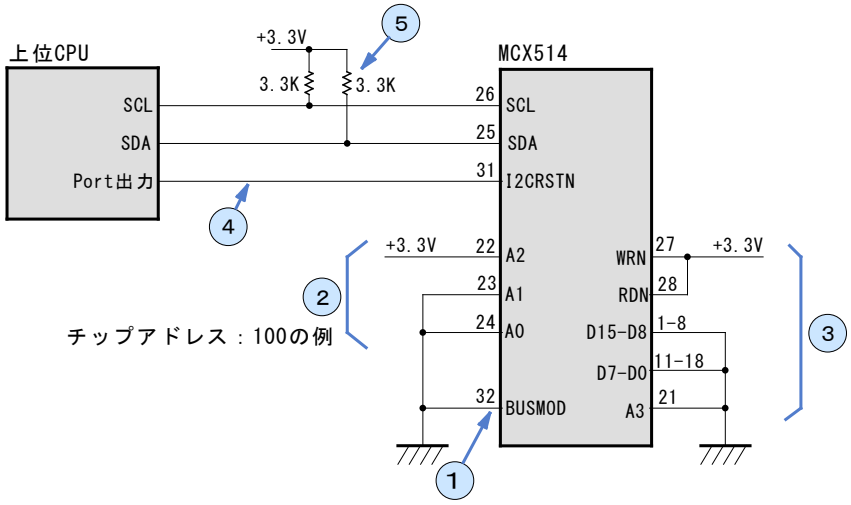
■ ストップコンデション生成
データ読み出しを終了させる時にはストップコンデションを生成します。SCL 信号が Hi のときに SDA 信号が Low から Hi に変化するとストップコンデションとなります。いかなる送受信においても CPU 側が最後にこのコンデションを作る必要があります。

4.2.3 I²C シリアルバス使用時の注意点

- WR0 レジスタ書き込み時は、必ず上位バイト(H)を先に、下位バイト(L)を後から書き込むようにしてください。連続して2バイト書き込むことができません。それぞれ個別にスレーブアドレスをセットして書き込む必要があります。下位バイトを書き込むと、先に指定された軸に対して、直ちに命令が実行されます。
- データ読み出し時、最終データを読み出した時の ACK 信号は、CPU 側は Low レベルではなく Hi レベルの ACK 信号を SDA ライン上に返すようにしてください。Low レベルを返すと、本 IC は正常に通信を終了できません。
- INT0N 信号関連の割り込みを使用する場合、RR0H の読み出しはできません。RR0H を読み出した場合、INT0N 信号関連の割り込みがクリアされてしまう可能性があります。割り込み使用時に RR0H の読み出しも行いたい場合は、弊社までお問い合わせください。
- RR1 レジスタを読み出す時は、必ず RR1L から 2 バイト(RR1L、RR1H)読み出すようにしてください。RR1L の 1 バイトだけを読み出した場合、RR1H の割り込みもクリアされてしまう可能性があります。
- リピートスタートコンデションは使用できません。

4.2.4 接続例

本 IC と CPU との接続例を以下に示します。



① BUSMOD を GND に固定し、I²C バスモードにします。
② A2,A1,A0 信号でチップアドレスを決めます。
③ パラレルバス用信号(フローティング入力)は GND または VCC に固定します。
④ I2CRSTN は、必要ならば配線します。初期設定時に SCL,SDA ラインに CPU 側からノイズが発生しなければ必要ありません。
⑤ SCL,SDA ラインには必ずプルアップ抵抗が必要です。

4.2.5 制御例

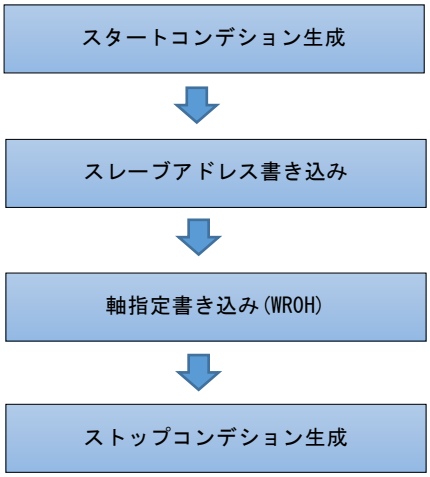
I²C シリアルバスを使用した本 IC の制御の流れを、次の 3 つの例で示します。また、これらの 3 つの制御を含む各種 CPU との制御サンプルプログラムを弊社ホームページ（http://www.novaelec.co.jp/）からダウンロードできます。

(1) 命令発行
(2) データ書き込み
(3) データ読み出し

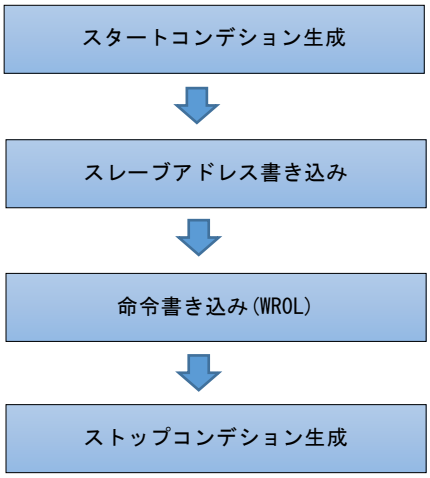
(1) 命令発行

命令を発行するには、命令を実行する軸と命令をWR0レジスタに書き込みます。このとき、WR0Lの下位バイトを書き込むと直ちに命令が実行されますので、命令発行前に軸指定をする必要があります。下記の通り、①で軸指定をした後に②で命令を書き込むようにします。

① WR0H へ軸指定書き込み



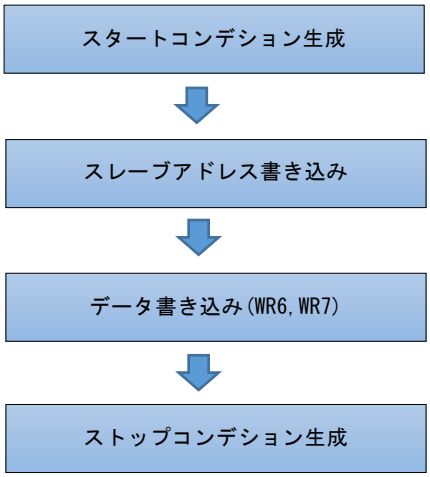
② WR0L へ命令書き込み



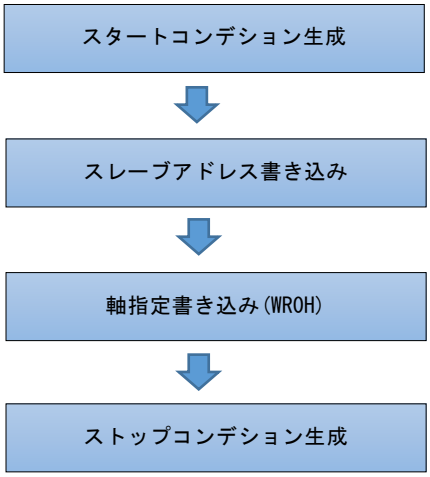
(2) データ書き込み

パラメータ設定などのデータ書き込みは、パラメータ値をWR6、WR7レジスタに書き込み、その後、WR0レジスタに軸指定と命令を発行することで行います。WR0 レジスタへの書き込みは、「(1) 命令発行」で記したように、軸指定後に命令を発行する必要があります。

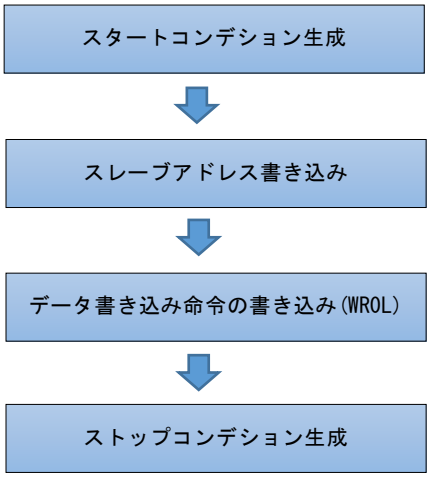
① WR6,7 へデータ書き込み



② WR0H へ軸指定書き込み



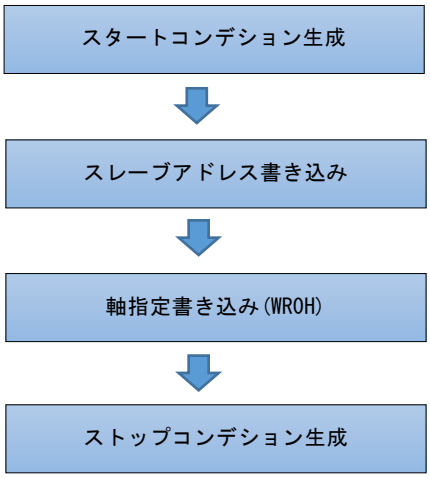
③ WR0L へデータ書き込み命令の書き込み



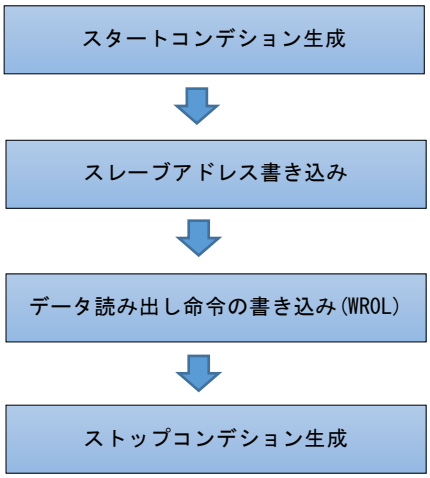
(3) データ読み出し

データ読み出しは、WR0 レジスタに軸指定と命令を発行後に、RR6、RR7 レジスタを読み出すことで行います。WR0 レジスタへの書き込みは、「(1) 命令発行」で記したように、軸指定後に命令を発行する必要があります。

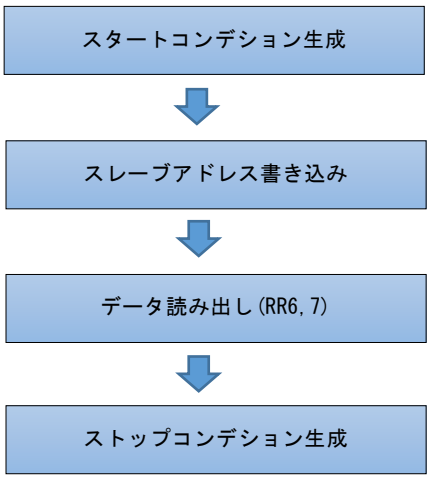
① WR0H へ軸指定書き込み



② WR0L へデータ読み出し命令の書き込み

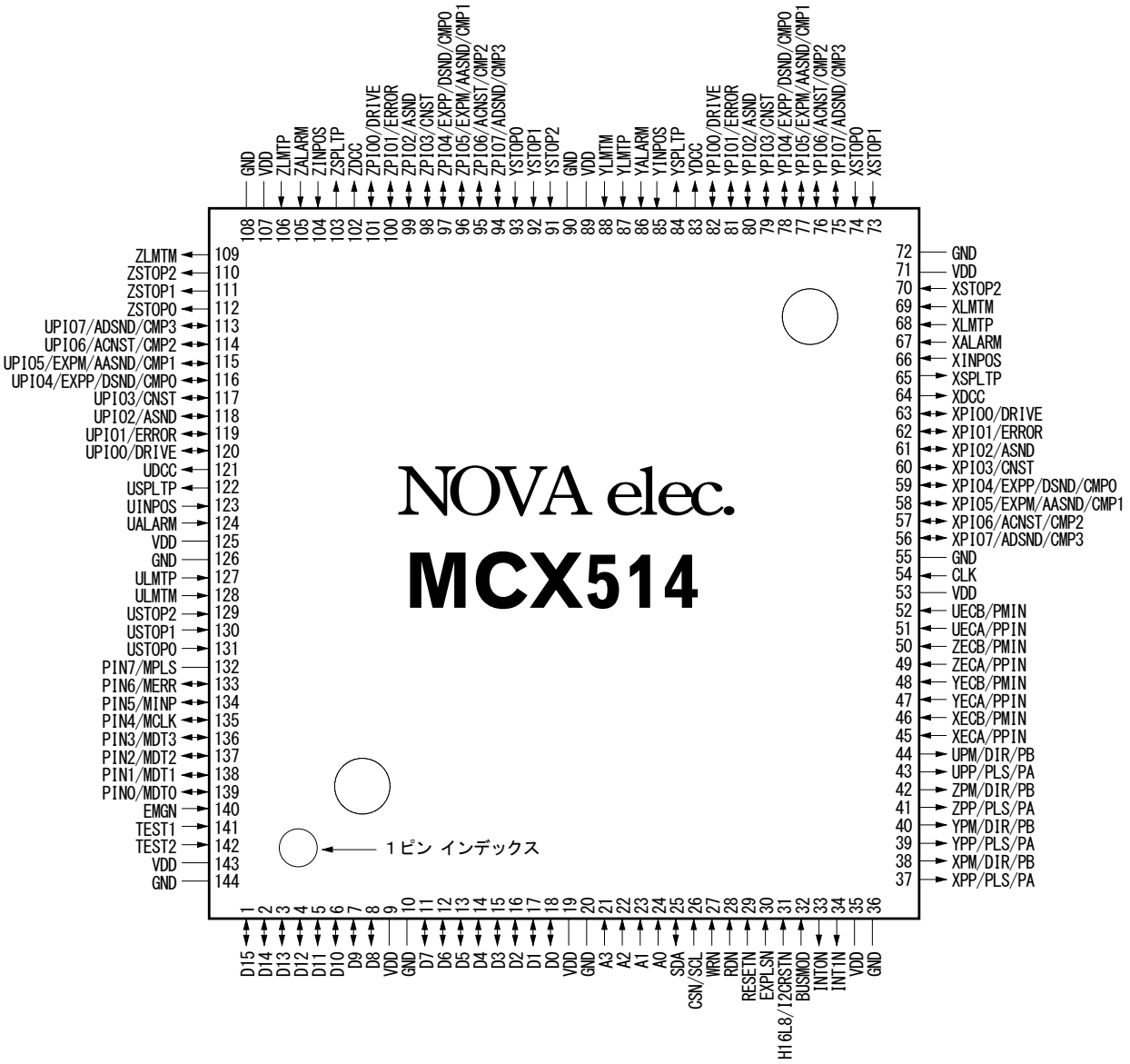


③ RR6,7 の読み出し



5. 端子配置と各信号の説明

5.1 端子配置



144 ピン プラスチック QFP
パッケージ外形 20×20mm, 最外形 22×22mm, 端子ピッチ 0.5mm
外形寸法は 10 章に記載されています。

5.2 各信号の説明

入／出力回路は5.3節に説明されていますので参照してください。－Ｆ－記号の付いた入力信号は本IC内部入力段に積分フィルタ回路を持っています。

信号名	端子番号	入／出力回路	信号の説明
CLK	54	入力Ａ	Clock：本ＩＣの内部同期回路を動作させるクロック信号です。周波数16.000MHzのクロックを入力します。ドライブ速度、加／減速度、加／減速度増加率はこのクロックの周波数に依存します。16MHz以外の周波数を入力する場合は速度設定値、加減速設定値などが異なってきます。
D15～D0	1～8, 11～18	双方向Ａ	Data Bus (D15～D0)：３ステート双方向の１６ビットデータバスです。システムのデータバスに接続します。CSN＝Lowで RDN＝Low のとき出力状態になります。これ以外のときはハイインピーダンスの入力状態になっています。 データバスを８ビットで使用する場合で、D15～D8を使用しない場合は、高抵抗（10k～100kΩ程度）を介して VDD または GND に接続してください。 I^2C モードでは、汎用入力信号として使用できます。
A3～A0	21～24	入力Ａ	Address：上位ＣＰＵが本ＩＣのリード／ライトレジスタを選択するためのアドレス信号です。 データバスを１６ビットで使用する場合は、A3は使用しませんので、GNDに接続してください。 I^2C モードでは、A2～A0はチップアドレス設定端子として使用します。
SDA	25	双方向Ｄ	I2CSDA：I^2C モード時の SDA 信号です。
CSN/SCL	26	入力Ａ	Chip Select / I^2C SCL：本ＩＣをＩ／Ｏデバイスとして選択するための入力信号です。本ＩＣをリード／ライトアクセスするとき、Low レベルにします。 I^2C モードでは SCL 信号として使用します。
WRN	27	入力Ａ	Write Strobe：本ＩＣのライトレジスタに書き込みを行うときに Low にします。WRN が Low の期間は CSN および A3～A0 が確定していなければなりません。WRN が↑のとき、データバスの内容がライトレジスタにラッチされるので、WRN の↑の前後は D15～D0 の値が確定していなければなりません。
RDN	28	入力Ａ	Read Strobe：本ＩＣのリードレジスタからデータを読み出すときに Low にします。CSN を Low にし RDN を Low にすると、RDN が Low の期間だけ、A3～A0 のアドレス信号によって選択されたリードレジスタのデータがデータバスに出力されます。
RESETN	29	入力Ａ	Reset：本ＩＣをリセット（初期化）する信号です。CLKが８サイクル以上の間RESETNをLow にするとリセットされます。電源投入時には、必ず本ＩＣをRESETN信号でリセットしなければなりません。 【注意】CLK が入力されていないと RESETN を Low にしてもリセットされません。
EXPLSN	30	入力Ｂ	External Pulse：補間ドライブ外部信号ステップ送りモードのときのパルス入力です。補間ドライブ外部信号ステップ送りモードの補間ドライブでは、EXPLSN の↓で１パルス分の補間演算が起動し、各軸の補間パルスが１パルス出力されます。 EXPLSN の Low レベルパルス幅は、最小４CLK 以上必要です。 【注意】EXPLSN はフィルタ機能がありません
H16L8 /I2CRSTN	31	入力Ｂ	Hi=16bit,Low=8bit：１６ビットデータバス／８ビットデータバスを選択します。Hi レベルにすると１６ビットデータバスになりＩＣ内のリードライトレジスタを１６ビットでアクセスします。また、Low レベルにすると、データバスは D7～D0 の８ビットのみ有効となり、内部リード／ライトレジスタを８ビットでアクセスします。 I^2C モードでは I^2C リセットとして使用します。Low にすると、IC 内部の I^2C 制御部のみリセットされます。
BUSMOD	32	入力Ｂ	Bus Mode：CPU バスモードを選択します。Hi レベルにすると１６ビット／８ビットパラレルバスモードになります。Low レベルにすると、I^2C シリアルバスモードになります。
INTON	33	出力Ｂ	Interrupt：上位ＣＰＵに対する割り込み要求信号です。補間プリバッファ以外のいずれかの割り込み要因により割り込みが発生すると INTON は Low レベルになります。割り込みが解除されると、Hi-Z に戻ります。

信号名	端子番号	入／出力回路	信号の説明
INT1N	34	出力Ｂ	Interrupt：上位ＣＰＵに対する割り込み要求信号です。補間プリバッファの割り込み要因により割り込みが発生すると INT1N は Low レベルになります。割り込みが解除されると、Hi-Z に戻ります。
XPP/PLS/PA YPP/PLS/PA ZPP/PLS/PA UPP/PLS/PA	37 39 41 43	出力Ａ	Pulse + / Pulse / Pulse Phase A：＋方向のドライブパルスを出力します。リセット時の状態はLowレベルになっており、ドライブ動作に入ると、デューティ５０％（定速時）の正パルスが出力されます。 モード選択で１パルス方式が選択された場合には、本端子よりドライブパルスが出力されます。 モード選択で２相パルス方式が選択された場合には、本端子よりＡ相信号が出力されます。
XPM/DIR/PB YPM/DIR/PB ZPM/DIR/PB UPM/DIR/PB	38 40 42 44	出力Ａ	Pulse - / Direction / Pulse Phase B：－方向のドライブパルスを出力します。リセット時の状態はLowレベルになっており、ドライブ動作に入ると、デューティ５０％（定速時）の正パルスが出力されます。 モード選択で１パルス方式が選択された場合には、本端子は方向信号となります。 モード選択で２相パルス方式が選択された場合には、本端子よりＢ相信号が出力されます。
XECA/PPIN YECA/PPIN ZECA/PPIN UECA/PPIN	45 47 49 51	入力Ｂ －Ｆ－	Encoder-A / Pulse+in：エンコーダＡ相信号の入力です。Ｂ相信号とともに、ＩＣ内部でアップ／ダウンパルスに変換され、実位置カウンタのカウント入力になります。 モード選択をアップ／ダウンパルス入力に選択すると、本端子はアップパルス入力となり、入力パルスの↑で、実位置カウンタがカウントアップされます。
XECB/PMIN YECB/PMIN ZECB/PMIN UECB/PMIN	46 48 50 52	入力Ｂ －Ｆ－	Encoder-B / Pulse-in：エンコーダＢ相信号の入力です。Ａ相信号とともに、ＩＣ内部でアップ／ダウンパルスに変換され、実位置カウンタのカウント入力になります。 モード選択をアップ／ダウンパルス入力に選択すると、本端子はダウンパルス入力となり、入力パルスの↑で、実位置カウンタがカウントダウンされます。
XSTOP2～0 YSTOP2～0 ZSTOP2～0 USTOP2～0	70,73,74 91,92,93 110,111,112 129,130,131	入力Ｂ －Ｆ－	Stop2～0：ドライブを途中で減速停止または即停止させるための入力信号です。サーチ動作の入力信号として使用します。フィルタ機能が無効の場合、2CLK以上のアクティブパルス幅が必要です。STOP2～STOP0それぞれについて有効／無効、論理レベルを設定することができます。 自動原点出しでは、STOP0を原点近傍信号に、STOP1を原点信号に、STOP2をエンコーダＺ相信号に割り当てることができます。 また、これらの信号状態はRR3レジスタ ページ0で常時読み出すことができます。
XLMTP YLMTP ZLMTP ULMTP	68 87 106 127	入力Ｂ －Ｆ－	Over Run Limit +：＋方向のオーバーランリミット信号です。＋方向のドライブパルス出力中に、この信号がアクティブになるとドライブは減速停止または即停止します。フィルタ機能が無効の場合、2CLK以上のアクティブパルス幅が必要です。有効／無効、減速停止／即停止、論理レベルをモード選択することができます。 リミット信号を有効にして、＋方向ドライブ中にこの信号がアクティブレベルになると、RR2レジスタのHLMT+ビットに１が立ちます。 この信号状態はRR3レジスタ ページ0で常時読み出すことができます。 また、この信号を自動原点出しの検出信号として割り当てることができます。
XLMTM YLMTM ZLMTM ULMTM	69 88 109 128	入力Ｂ －Ｆ－	Over Run Limit -：－方向のオーバーランリミット信号です。－方向のドライブパルス出力中に、この信号がアクティブになるとドライブは減速停止または即停止します。フィルタ機能が無効の場合、2CLK以上のアクティブパルス幅が必要です。有効／無効、減速停止／即停止、論理レベルをモード選択することができます。 リミット信号を有効にして、－方向ドライブ中にこの信号がアクティブレベルになると、RR2レジスタのHLMT-ビットに１が立ちます。 この信号状態はRR3レジスタ ページ0で常時読み出すことができます。 また、この信号を自動原点出しの検出信号として割り当てることができます。

信号名	端子番号	入／出力回路	信号の説明
XINPOS YINPOS ZINPOS UINPOS	66 85 104 123	入力Ｂ －Ｆ－	Inposition：サーボモータドライバのインポジション（位置決め完了）出力に対応する入力信号です。有効／無効、論理レベルはモード選択することができます。有効に設定すると、ドライブ終了後、この信号がアクティブになるのを待ってから、主ステータスレジスタのDRIVEビットが０に戻ります。 この信号状態はRR3レジスタ ページ0で常時読み出すことができます。
XALARM YALARM ZALARM UALARM	67 86 105 124	入力Ｂ －Ｆ－	Servo Alarm：サーボモータドライバのアラーム出力に対応する入力信号です。有効／無効、論理レベルはモード選択することができます。有効に設定すると、ドライブ中にこの信号がアクティブレベルに変化するとRR2レジスタのALARMビットに１が立ち、ドライブは即停止します。 この信号状態はRR3レジスタ ページ0で常時読み出すことができます。
XPIO7/ADSND/CMP3 YPIO7/ADSND/CMP3 ZPIO7/ADSND/CMP3 UPIO7/ADSND/CMP3	56 75 94 113	双方向Ｂ －Ｆ－	Universal Input Output7 / Acceleration Descend / Compare MR3：汎用入出力信号(PIO7)、加速度減少状態出力信号(ADSND)、MR3 比較出力(CMP3)の端子を共用しています。使用する信号をモード選択できます。 汎用入力信号(PIO7)の状態は、XPIO7,YPIO7 は RR4 レジスタ、ZPIO7,UPIO7 は RR5 レジスタで常時読み出すことができます。 汎用出力信号(PIO7)は XPIO7,YPIO7 は WR4 レジスタ、ZPIO7,UPIO7 は WR5 レジスタに 1/0 データを書き込むことによって Hi/Low にします。 同期動作では、起動要因の入力信号として使用できます。 加速度減少状態出力(ADSND)は、ドライブ命令実行中、加速度減少状態になると Hi になります。 MR3 比較出力(CMP3)は多目的レジスタ MR3 の比較条件を満たしているとき Hi になります。
XPIO6/ACNST/CMP2 YPIO6/ACNST/CMP2 ZPIO6/ACNST/CMP2 UPIO6/ACNST/CMP2	57 76 95 114	双方向Ｂ －Ｆ－	Universal Input Output6 / Acceleration Constant / Compare MR2：汎用入出力信号(PIO6)、加速度一定状態出力信号(ACNST)、MR2 比較出力(CMP2)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO6)については、PIO7 と同様です。 同期動作では、起動要因の入力信号として使用できます。 加速度一定状態出力(ACNST)は、ドライブ命令実行中、加速度一定状態になると Hi になります。 MR2 比較出力(CMP2)は多目的レジスタ MR2 の比較条件を満たしているとき Hi になります。
XPIO5/EXPM/AASND/CMP1 YPIO5/EXPM/AASND/CMP1 ZPIO5/EXPM/AASND/CMP1 UPIO5/EXPM/AASND/CMP1	58 77 96 115	双方向Ｂ －Ｆ－	Universal Input Output5 / External Operation- / Acceleration Ascend / Compare MR1：汎用入出力信号(PIO5)、外部操作入力(EXPM)、加速度増加状態出力信号(AASND)、MR1 比較出力(CMP1)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO5)については、PIO7 と同様です。 同期動作では、起動要因の入力信号として使用できます。 外部操作入力(EXPM)は、外部から－方向のドライブを起動する信号です。外部相対位置ドライブモードにすると、本信号の↓で－方向の相対位置ドライブが起動します。外部連続パルスドライブモードにすると、本信号が Low レベルの間、連続して－方向連続パルスドライブが行われます。手動パルサーモードの場合は、エンコーダＢ相信号を本端子に入力します。 加速度増加状態出力(AASND)は、ドライブ命令実行中、加速度増加状態になると Hi になります。 MR1 比較出力(CMP1)は多目的レジスタ MR1 の比較条件を満たしているとき Hi になります。
XPIO4/EXPP/DSND/CMP0 YPIO4/EXPP/DSND/CMP0 ZPIO4/EXPP/DSND/CMP0 UPIO4/EXPP/DSND/CMP0	59 78 97 116	双方向Ｂ －Ｆ－	Universal Input Output4 / External Operation+ / Descend / Compare MR0：汎用入出力信号(PIO4) 、外部操作入力(EXPP)、減速状態出力信号(DSND)、MR0 比較出力(CMP0)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO4)については、PIO7 と同様です。 同期動作では、起動要因の入力信号として使用できます。 外部操作入力(EXPM)は、外部から＋方向のドライブを起動する信号です。外部相対位置ドライブモードにすると、本信号の↓で＋方向の相対位置ドライブが起動します。外部連続パルスドライブモードにすると、本信号が Low レベルの間、＋方向連続パルスドライブが行われます。手動パルサーモードの場合は、エンコーダＡ相信号を本端子に入力します。 減速状態出力信号(DSND)は、ドライブ命令実行中、減速状態になると Hi になります。 MR0 比較出力(CMP0)は多目的レジスタ MR0 の比較条件を満たしているとき Hi になります。

信号名	端子番号	入／出力回路	信号の説明
XPIO3/CNST YPIO3/CNST ZPIO3/CNST UPIO3/CNST	60 79 98 117	双方向Ｂ －Ｆ－	Universal Input Output3 / Constant：汎用入出力信号(PIO3)、定速状態出力信号(CNST)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO3)については、PIO7 と同様です。 同期動作では、起動要因の入力信号として、または動作の同期パルスの出力信号として使用できます。同期パルスの論理レベル、パルス幅はモード選択できます。 定速状態出力信号(CNST)は、ドライブ命令実行中、定速状態になると Hi になります。
XPIO2/ASND YPIO2/ASND ZPIO2/ASND UPIO2/ASND	61 80 99 118	双方向Ｂ －Ｆ－	Universal Input Output2 / Ascend：汎用入出力信号(PIO2)、加速状態出力信号(ASND)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO2)については、PIO7 と同様です。 同期動作については、PIO3 と同様です。 加速状態出力信号(ASND)は、ドライブ命令実行中、加速状態になると Hi になります。
XPIO1/ERROR YPIO1/ERROR ZPIO1/ERROR UPIO1/ERROR	62 81 100 119	双方向Ｂ －Ｆ－	Universal Input Output1 / Error：汎用入出力信号(PIO1)、エラー状態出力信号(ERROR)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO1)については、PIO7 と同様です。 同期動作については、PIO3 と同様です。 エラー状態出力(ERROR)はエラーが発生している期間､Hi レベルになります。
XPIO0/DRIVE YPIO0/DRIVE ZPIO0/DRIVE UPIO0/DRIVE	63 82 101 120	双方向Ｂ －Ｆ－	Universal Input Output0 / Drive：汎用入出力信号(PIO0)、ドライブ状態出力信号(DRIVE)の端子を共用しています。使用する信号をモード選択できます。 汎用入出力信号(PIO0)については、PIO7 と同様です。 同期動作については、PIO3 と同様です。 ドライブ状態表示出力(DRIVE)は、ドライブパルスを出力している期間、Hi レベルになります。自動原点出し実行時には、実行している間、本信号が Hi レベルになります。モード選択で、サーボモータ用の INPOS 信号を有効にしている場合は、INPOS がアクティブになるまで、DRIVE 信号は Hi になっています。
XDCC YDCC ZDCC UDCC	64 83 102 121	出力Ａ	Deviation Counter Clear：偏差カウンタクリア出力信号です。サーボモータドライバに対して出力する信号です。自動原点出しでモード設定することにより出力させることができます。また、コマンドにより出力させることができます。
XSPLTP YSPLTP ZSPLTP USPLTP	65 84 103 122	出力Ａ	Split Pulse：スプリットパルスを出力します。スプリットパルスの出力開始／停止は、同期動作およびコマンドによって行うことができます。 スプリット長、パルス幅、パルス数をコマンドにより設定することができます。 また、出力論理、開始パルスあり／なしをモード選択することができます。
PIN7/MPLS	132	双方向Ｃ	Universal Input7/：汎用入力信号です。汎用入力値読み出し命令(48h)で入力値を取得できます。Lowレベルが０、Hiレベルが１です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3 kΩの抵抗でVDD(+3.3V)にプルアップします。
PIN6/MERR	133	双方向Ｅ	Universal Input6/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3 kΩの抵抗でVDD(+3.3V)にプルアップします。
PIN5/MINP	134	双方向Ｅ	Universal Input5/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3 kΩの抵抗でVDD(+3.3V)にプルアップします。
PIN4/MCLK	135	双方向Ｃ	Universal Input4/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3 kΩの抵抗でVDD(+3.3V)にプルアップします。
PIN3/MDT3	136	双方向Ｃ	Universal Input3/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3 kΩの抵抗でVDD(+3.3V)にプルアップします。

信号名	端子番号	入／出力回路	信号の説明
PIN2/MDT2	137	双方向Ｃ	Universal Input2/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3kΩの抵抗でVDD(+3.3V)にプルアップします。
PIN1/MDT1	138	双方向Ｃ	Universal Input1/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3kΩの抵抗でVDD(+3.3V)にプルアップします。
PIN0/MDT0	139	双方向Ｃ	Universal Input0/：汎用入力信号です。読み出しはPIN7と同様です。 マルチチップ多軸補間を行うときは、この信号をチップ間で接続し、3.3kΩの抵抗でVDD(+3.3V)にプルアップします。
EMGN	140	入力Ｂ －Ｆ－	Emergency Stop：全軸のドライブを緊急停止させる入力信号です。ドライブ中にこの信号をLowレベルにすると、補間ドライブも含め、全軸のドライブが即停止し、各軸のRR2レジスタのEMGビットに１が立ちます。フィルタ機能が無効の場合、2CLK以上のLowレベルパルス幅が必要です。 【注意】この信号は、論理レベルを選択することはできません。
TEST1 TEST2	141,142	－	Test：内部回路の動作テストを行うための入力端子です。Hi にすると、ＩＣ内部のテスト回路が動作し、思わぬ動きをします。両端子ともに、必ずオープンか GND に接続してください。 ＩＣ内部で５０kΩで GND にプルダウンされています。
GND	10,20,36, 55,72,90, 108,126,144		グランド（０Ｖ）端子です。必ずすべての端子を接続してください。
VDD	9,19,35, 53,71,89, 107,125,143		電源端子です。＋3.3Vを供給してください。必ず、すべての端子を接続してください。

5.3 入／出力回路

入力Ａ	LVTTL レベルのシュミットトリガ入力です。ＩＣ内部では高抵抗でプルアップされておらず、ハイインピーダンスです。 本入力は、5V トレラントです。3.3V 系出力、および 5V 系出力(CMOS レベル、TTL レベル)のいずれの出力とも接続が可能です。 使用しない場合は、必ず GND または VDD に接続してください。
入力Ｂ	LVTTL レベルのシュミットトリガ入力です。ＩＣ内部では 50KΩでプルアップされています。 本入力は、5V トレラントです。3.3V 系出力、および 5V 系出力(CMOS レベル、TTL レベル)のいずれの出力とも接続が可能です。 使用しない場合は、オープンまたは VDD に接続してください。 －Ｆ－記号の付いた信号は、本ＩＣ内部入力段に積分フィルタ回路を持っています。
出力Ａ	3.3V 系 CMOS レベルの出力です。6mA 駆動バッファ(Hi レベル出力電流 I_{OH}=-6mA で V_{OH}=2.6Vmin、Low レベル出力電流 I_{OL}=6mA で V_{OL}=0.4Vmax)です。 Hi レベル出力時に外部より、出力電圧以上の電圧を印加しないで下さい。 5V 系入力との接続は、相手入力が TTL レベルであれば接続が可能です。相手入力が 5V 系 CMOS レベルの場合は接続することができません。（*注 1）
出力Ｂ	オープンドレイン出力です。12mA 駆動バッファ(Low レベル出力電流 I_{OL}=12mA で V_{OL}=0.4Vmax)です。 使用する場合は、高抵抗で VDD(+3.3V)にプルアップしてください。TTL レベルの 5V 系 IC への接続も可能です。
双方向Ａ	入力側は、5V トレラントの LVTTL レベルのシュミットトリガ入力です。ＩＣ内部では高抵抗でプルアップされておらず、ハイインピーダンスです。データ信号は、信号ラインがハイインピーダンスにならないよう、システム全体で高抵抗でプルアップしてください。 D15～D8､PIN6,5 を使用しないときは､高抵抗(10k～100kΩ程度)を介して VDD(+3.3V)または GND に接続してください。 双方向ですので、直接プルアップ／プルダウンするより、高抵抗を入れた方が無難です。 Hi レベル出力時に外部より、出力電圧以上の電圧を印加しないで下さい。 出力側は、3.3V 系 CMOS レベルの出力です。12mA 駆動バッファ(Hi レベル出力電流 I_{OH}=-12mA で V_{OH}=2.6Vmin、Low レベル出力電流 I_{OL}=12mA で V_{OL}=0.4Vmax)です。 5V 系双方向 IC との接続は、相手側入力が TTL レベルであれば接続が可能です。相手入力が 5V 系 CMOS レベルの場合は接続することができません。(*注 1)
双方向Ｂ	入力側は、5V トレラントの LVTTL レベルのシュミットトリガ入力です。ＩＣ内部では 50KΩでプルアップされています。 Hi レベル出力時に外部より、出力電圧以上の電圧を印加しないで下さい。 出力側は、3.3V 系 CMOS レベルの出力です。6mA 駆動バッファ(Hi レベル出力電流 I_{OH}=-6mA で V_{OH}=2.6Vmin、Low レベル出力電流 I_{OL}=6mA で V_{OL}=0.4Vmax)です。 5V 系双方向 IC との接続は、相手側入力が TTL レベルであれば接続が可能です。相手入力が 5V 系 CMOS レベルの場合は接続することができません。(*注 1) －Ｆ－記号の付いた信号は、本ＩＣ内部入力段に積分フィルタ回路を持っています。
双方向Ｃ	入力側は、5V トレラントの LVTTL レベルのシュミットトリガ入力です。ＩＣ内部では 100KΩでプルアップされています。 マルチチップ多軸補間を行うとき出力側が作動します。マルチチップ多軸補間で､チップ間で信号を接続するときは、配線をできるだけ短くし、他の信号線と交錯しないようにしてください。 使用しない場合はオープンで構いません。
双方向Ｄ	I^2C の SDA 信号専用です。 入力側は5V トレラントのLVTTL レベルのシュミットトリガ入力です。ＩＣ内部では高抵抗でプルアップされておらず、ハイインピーダンスです。 出力側は、6mA 駆動バッファのオープンドレイン出力です。SDA 信号として使用する場合は外部に抵抗を介して VDD にプルアップします。 使用しない場合は、高抵抗で VDD にプルアップするか GND に直接接続してください。
双方向Ｅ	入力側は5V トレラントのLVTTL レベルのシュミットトリガ入力です。ＩＣ内部では高抵抗でプルアップされておらず、ハイインピーダンスです。 出力側はオープンドレイン出力です。 マルチチップ多軸補間を行うとき出力側が作動します。チップ間で信号を接続するときは、配線をできるだけ短くし、他の信号線と交錯しないようにしてください。 使用しない場合は、高抵抗で VDD にプルアップするか GND に直接接続してください。

注 1: 出力 A および双方向 A,B の出力信号を外部で抵抗を介して 5V にプルアップしても､Hi レベル出力電圧を 5V 系 CMOS の Hi レベル入力電圧まで上げることはできません｡このような回路構成は行わないでください。

5.4 回路設計上の注意

a. TEST1,2 端子の処置

TEST1,2(141,142)端子は必ず GND に接続してください。Hi レベルにすると内部のテスト回路が作動し、正常な動作が全くできなくなります。

b. 未使用入力端子の処置

使用しない入力A端子は必ず GND または VDD に接続してください。使用しない入力端子をオープン状態にしておくと、端子の信号レベルが不安定になり IC の機能不良の原因となります。入力 B 端子はオープンで構いません。

c. 未使用双方向端子の処置

双方向端子(双方向A,D,E)の使用しない端子については、高抵抗(10k～100kΩ 程度)を介して VDD または GND に接続してください。直接 GND または VDD に接続すると、万一プログラムの誤りなどで出力状態になった場合には過電流が流れ IC が破損する場合があります。双方向 B,C 端子はオープンで構いません。

d. デカップリングコンデンサ

本ICの VDD と GND 間に、高周波特性の良い 0.1μF 程度のデカップリングコンデンサを 2～3 個入れてください。

e. 端子インダクタンスによるリンギングノイズ

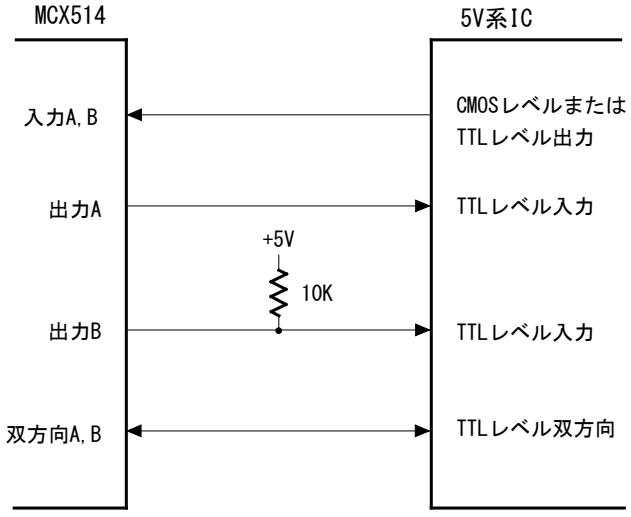
出力端子のもつインダクタンスと出力に接続される負荷容量の共振によって、出力信号の立ち上がり、立ち下がりでリンギングノイズが発生する場合があります。接続する次段の回路が誤動作するほどリンギングノイズが大きい場合には、10～100pF 程度の負荷容量を接続して、リンギングをおさえることができます。

f. 伝送路の反射

出力A, Bおよび双方向A～E タイプの出力時は、負荷容量を 20～50pF とした場合、信号の立ち上がり／立ち下がり時間が約3～4nsec になりますので、配線の長さが 60cm くらいから、反射の影響が著しくなってきます。配線路の長さは、できるだけ短くしてください。

g. 5V 系ICとの接続例

本ICの入／出力回路は 5V トレラントですが、出力回路は TTL レベルの入力との接続のみ可能です。CMOS レベルの入力との接続はできません



6. リード／ライトレジスタ

この章では、CPUが各軸を制御するためにアクセスするリード／ライトレジスタについて、詳細に記述します。

6.1 16 ビットデータバスのレジスタアドレス

下表に示すように、16ビットデータバスを使用する場合は、16ビットのリード／ライトレジスタにアクセスするためのアドレスが8あります。

■ 16ビットデータバスにおけるライトレジスタ

すべてのレジスタは 16 ビット長です。

アドレス A2 A1 A0	レジスタ記号	レジスタ名	内　　容
0 0 0	WR0	コマンドレジスタ	・軸指定、命令コードのセット
0 0 1	XWR1	Ｘ軸モードレジスタ１	・割り込みの許可／禁止の設定
	YWR1	Ｙ軸モードレジスタ１	
	ZWR1	Ｚ軸モードレジスタ１	
	UWR1	Ｕ軸モードレジスタ１	
0 1 0	XWR2	Ｘ軸モードレジスタ２	・外部減速停止信号の論理レベル、有効／無効の設定
	YWR2	Ｙ軸モードレジスタ２	・サーボモータ用信号の論理レベル、有効／無効の設定
	ZWR2	Ｚ軸モードレジスタ２	・リミット信号のモード設定、ソフトリミットのモード設定
	UWR2	Ｕ軸モードレジスタ２	
0 1 1	XWR3	Ｘ軸モードレジスタ３	・自動減速／マニュアル減速の設定
	YWR3	Ｙ軸モードレジスタ３	・加減速のモード設定（対称／非対称、直線加減速／Ｓ字加減速）
	ZWR3	Ｚ軸モードレジスタ３	・ドライブパルス出力のモード、端子設定
	UWR3	Ｕ軸モードレジスタ３	・エンコーダ入力信号のモード、端子設定
1 0 0	WR4	アウトプットレジスタ１	・Ｘ軸汎用入出力信号 XPIO7～0 出力値設定
			・Ｙ軸汎用入出力信号 YPIO7～0 出力値設定
1 0 1	WR5	アウトプットレジスタ２	・Ｚ軸汎用入出力信号 ZPIO7～0 出力値設定
			・Ｕ軸汎用入出力信号 UPIO7～0 出力値設定
1 1 0	WR6	ライトデータレジスタ１	・ライトデータ下位１６ビット（D15～D0）のセット
1 1 1	WR7	ライトデータレジスタ２	・ライトデータ上位１６ビット（D31～D16）のセット

● 上表で示すように、各軸とも、WR1、WR2、WR3(モードレジスタ 1,2,3)を持っています。これらのレジスタへは、同一アドレスで書き込みを行うことになります。どの軸のモードレジスタに書き込むかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したNOP命令を直前に書き込むことによって、書き込みたい軸を選択します。

● リセット時は、WR1,WR2,WR3,WR4,WR5 レジスタはすべてのビットが 0 にクリアされます。

■ 16ビットデータバスにおけるリードレジスタ

すべてのレジスタは16ビット長です。

アドレス A2 A1 A0	レジスタ記号	レジスタ名	内　　容
0 0 0	RR0	主ステータスレジスタ	・ドライブ状態、エラー状態を表示 ・連続補間次データ可、円弧補間の象限、連続補間プリバッファスタックカウンタ(SC)の表示
0 0 1	XRR1 YRR1 ZRR1 URR1	Ｘ軸ステータスレジスタ１ Ｙ軸ステータスレジスタ１ Ｚ軸ステータスレジスタ１ Ｕ軸ステータスレジスタ１	・割り込み発生要因の表示
0 1 0	XRR2 YRR2 ZRR2 URR2	Ｘ軸ステータスレジスタ２ Ｙ軸ステータスレジスタ２ Ｚ軸ステータスレジスタ２ Ｕ軸ステータスレジスタ２	・エラー発生要因の表示 ・終了ステータスの表示
0 1 1	XRR3 YRR3 ZRR3 URR3	Ｘ軸ステータスレジスタ３ Ｙ軸ステータスレジスタ３ Ｚ軸ステータスレジスタ３ Ｕ軸ステータスレジスタ３	● ページ０ ・入力信号の状態表示 ・自動原点出し実行ステートの表示 ● ページ１ ・同期動作セットの有効／無効状態を表示 ・加／減速状態、加／減速度の増加／減少状態を表示 ・タイマー作動、スプリットパルス作動の状態を表示 ・マルチチップ補間時の終点データ転送エラー表示
1 0 0	RR4	ＰＩＯリードレジスタ１	・Ｘ軸汎用入出力信号の状態表示 ・Ｙ軸汎用入出力信号の状態表示
1 0 1	RR5	ＰＩＯリードレジスタ２	・Ｚ軸汎用入出力信号の状態表示 ・Ｕ軸汎用入出力信号の状態表示
1 1 0	RR6	リードデータレジスタ１	・リードデータ下位１６ビット（D15～D0）の表示
1 1 1	RR7	リードデータレジスタ２	・リードデータ上位１６ビット（D31～D16）の表示

● ライトレジスタと同様に、各軸とも、RR1、RR2、RR3(各軸ステータスレジスタ 1,2,3)を持っています。これらのレジスタは、同一アドレスで読み出しを行うことになります。どの軸のステータスレジスタに読み出すかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したNOP命令を直前に書き込むことによって、読み出したい軸を選択します。

● RR3 レジスタに関しては、ページ 0 とページ 1 の 2 種類存在します。RR3 ページ表示命令(7Ah、7Bh)を書き込むことでページの指定をします。リセット時はページ 0 となります。

6.2 8 ビットデータバスのレジスタアドレス

8 ビットデータバスでアクセスする場合は、16 ビットレジスタを上位バイト、下位バイトに分けてアクセスします。
下表において、****L は 16 ビットレジスタ****の下位バイト(D7～D0)、****H は 16 ビットレジスタ****の上位バイト(D15～D8)を示しています。コマンドレジスタ(WR0L, WR0H)だけは、必ず上位バイト(WR0H)を先に、下位バイト(WR0L)を後から書き込みます。

■ 8ビットデータバスにおけるライトレジスタ

アドレス A3 A2 A1 A0	ライトするレジスタ
0 0 0 0	WR0L
0 0 0 1	WR0H
0 0 1 0	WR1L
0 0 1 1	WR1H
0 1 0 0	WR2L
0 1 0 1	WR2H
0 1 1 0	WR3L
0 1 1 1	WR3H
1 0 0 0	WR4L
1 0 0 1	WR4H
1 0 1 0	WR5L
1 0 1 1	WR5H
1 1 0 0	WR6L
1 1 0 1	WR6H
1 1 1 0	WR7L
1 1 1 1	WR7H

■ 8ビットデータバスにおけるリードレジスタ

アドレス A3 A2 A1 A0	リードするレジスタ
0 0 0 0	RR0L
0 0 0 1	RR0H
0 0 1 0	RR1L
0 0 1 1	RR1H
0 1 0 0	RR2L
0 1 0 1	RR2H
0 1 1 0	RR3L
0 1 1 1	RR3H
1 0 0 0	RR4L
1 0 0 1	RR4H
1 0 1 0	RR5L
1 0 1 1	RR5H
1 1 0 0	RR6L
1 1 0 1	RR6H
1 1 1 0	RR7L
1 1 1 1	RR7H

6.3 I^2C シリアルインターフェイスバスモードのレジスタアドレス

本 IC を I^2C シリアルインターフェイスバスで使用する場合、スレーブアドレス制御でレジスタアドレスを指定してアクセスします。レジスタアドレスの指定方法は 6.2 節に記した方法と同じで、16 ビットレジスタを上位バイト、下位バイトに分けて指定します。
I^2C シリアルインターフェイスバスに関しては、4 章を参照してください。

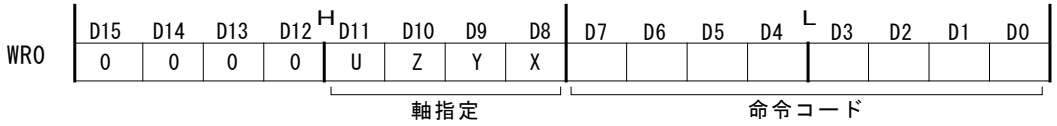
6.4 WR0 コマンドレジスタ

IC内の各軸に対して、軸指定をして、命令を書き込むレジスタです。レジスタは、軸を指定するビット、命令コードをセットするビットから成っています。

このレジスタに命令コードを書き込むと、その命令は直ちに実行されます。ドライブ速度の設定などのデータ書き込み命令は、あらかじめ、WR6,7 レジスタにデータが書き込まれていなければなりません。また、データ読出し命令は、このコマンドレジスタに命令を書き込むと、内部回路から RR6,7 レジスタにデータがセットされます。

8ビットデータバスのときは、必ず上位バイト(H)を先に、下位バイト(L)を後から書き込みます。下位バイトを書き込むと、先に指定された軸に対して、直ちに命令が実行されます。

すべての命令コードの命令処理に要する時間は、最大で 125nsec(CLK=16MHz の場合)です。この間は、次の命令を書き込まないでください。



D7～0　　命令コードをセットします。

D11～8　　命令を実行する軸を指定します。各軸のビットに1を立てるとその軸が指定されます。軸の指定は、1軸とは限りません。同時に複数の軸に対して同じ命令を発行したり、同じパラメータ値を書き込むことができます。ただし、データ読み出し命令の場合は1軸のみを指定してください。

その他のビットは必ず0にしてください。1にすると、IC内部回路のテスト命令が起動し、思わぬ動作をする場合があります。

6.5 WR1 モードレジスタ1

モードレジスタ１は４軸各々が個別に持っています。どの軸のモードレジスタに書き込むかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したＮＯＰ命令を直前に書き込むことによって、書き込みたい軸を選択します。

モードレジスタ1は各割り込み発生要因の許可／禁止を設定するレジスタです。各ビットは、1にすると割り込み許可、0にすると割り込み禁止になります。

D3～0　CMR3～0　多目的レジスタ MR3～0 と比較対象との比較結果が、比較条件を満たすように変化したとき、割り込みが発生します。
MR3～0 の比較対象および比較条件の設定は、多目的レジスタモード設定命令(20h)で行います。

D4　D-STA　ドライブが開始したとき、割り込みが発生します。

D5　C-STA　加減速ドライブ時に、定速域でのパルス出力を開始したとき、割り込みが発生します。

D6　C-END　加減速ドライブ時に、定速域でのパルス出力を終了したとき、割り込みが発生します。

D7　D-END　ドライブが終了したとき、割り込みが発生します。

D8　H-END　自動原点出しが終了したとき、割り込みが発生します。

D9　TIMER　タイマーがタイムアップしたとき、割り込みが発生します。

D10 SPLTP スプリットパルスごとのパルスの↑で割り込みが発生します。(スプリットパルス論理:Hi パルス設定時)

D11 SPLTE スプリットパルスが終了したとき、割り込みが発生します。

D15～12 SYNC3～0 同期動作 SYNC3～0 が起動したとき、割り込みが発生します。

リセット時には、D15～D0 は、すべて0にセットされます。

6.6 WR2 モードレジスタ2

モードレジスタ2は4軸各々が個別に持っています。どの軸のモードレジスタに書き込むかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したNOP命令を直前に書き込むことによって、書き込みたい軸を選択します。

モードレジスタ2は、ドライブ途中で減速停止／即停止させる入力信号 nSTOP2～nSTOP0 のモード設定、サーボモータ用入力信号のモード設定、リミット入力信号のモード設定、およびソフトリミットのモード設定を行うレジスタです。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR2	SLM-M	SLM-O	SLM-E	HLM-M	HLM-E	HLM-L	ALM-E	ALM-L	INP-E	INP-L	SP2-E	SP2-L	SP1-E	SP1-L	SP0-E	SP0-L

D4,2,0 SPk-L ドライブ停止入力信号 nSTOPk(k:2～0)の有効の論理レベルを設定するビットです。
0:Low でアクティブ、1:Hi でアクティブ
自動原点出しでは、使用する nSTOPk 信号の論理レベルを、これらのビットで設定します。

D5,3,1 SPk-E ドライブ停止入力信号 nSTOPk(k:2～0)の有効／無効を設定するビットです。
0:無効、1:有効
nSTOP2～nSTOP0 を有効にしてドライブを開始すると、指定の nSTOPk 信号入力がアクティブレベルになると、ドライブは減速停止または即停止します。加減速ドライブであれば減速停止、定速ドライブであれば即停止します。
自動原点出しでは、使用する nSTOPk 有効／無効ビットは0(無効)に設定しておきます。

D6 INP-L サーボモータ位置決め完了用入力信号 nINPOS の論理レベルを設定します。
0:Low でアクティブ、1:Hi でアクティブ

D7 INP-E nINPOS 入力信号の有効／無効を設定します。
0:無効、1:有効
有効に設定すると、ドライブ終了後、nINPOS 信号がアクティブになるのを待ってから RR0(主ステータス)レジスタの DRIVE ビットが0に戻ります。

D8 ALM-L サーボモータアラーム用入力信号 nALARM の論理レベルを設定します。
0:Low でアクティブ、1:Hi でアクティブ

D9 ALM-E nALARM 入力信号の有効／無効を設定します。
0:無効、1:有効
有効に設定すると、ドライブ中に nALARM 入力信号を監視し、アクティブ状態になると RR2 レジスタの D4(ALARM)ビットに1が立ちます。アクティブレベルになると、ドライブは即停止します。

D10 HLM-L ハードウェアリミット入力信号 nLMTP, nLMTM の論理レベルを設定します。
0:Low でアクティブ、1:Hi でアクティブ

D11 HLM-E nLMTP, nLMTM リミット入力信号の有効／無効を設定します。
0:無効、1:有効
有効に設定すると、＋方向ドライブ中に nLMTP リミット入力信号がアクティブ状態になると RR2 レジスタの D2(HLMT+)に1が立ち、－方向ドライブ中に nLMTM リミット入力信号がアクティブ状態になると、RR2 レジスタの D3(HLMT-)に1が立ちます。アクティブレベルになると、ドライブは停止します。

D12 HLM-M nLMTP, nLMTM リミット入力信号がアクティブになったときのドライブ停止方式を設定します。
0:即停止、1:減速停止
自動原点出しの停止信号にリミット信号を使用する場合は、1:減速停止に設定しておきます。

D13 SLM-E ソフトリミット機能の有効／無効を設定します。
0:無効、1:有効
有効に設定すると、＋方向ドライブ中に＋方向のソフトリミットエラー状態になると、RR2 レジスタの D0(SLMT+)に1が立ち、－方向ドライブ中に－方向のソフトリミットエラー状態になると、RR2 レジスタの D1(SLMT-)に1が立ちます。
・ ＋方向のソフトリミット:対象の位置カウンタ ≧ SLMT+値ならばエラー、ドライブ停止
・ －方向のソフトリミット:対象の位置カウンタ ＜ SLMT-値ならばエラー、ドライブ停止
ソフトリミットでエラーになっている方向のドライブ命令を書き込んでも、実行されません。

D14 SLM-O ソフトリミットの設定対象を論理位置カウンタにするか、実位置カウンタにするかを設定します。
0:論理位置カウンタ、 1:実位置カウンタ

D15 SLM-M ソフトリミット時のドライブ停止方式を設定します。
0:減速停止、1:即停止
(ハードウェアリミット信号の停止方式の設定と、ビットの0／1が逆であることに注意してください。)

リセット時には、D15～D0 は、すべて0にセットされます。

6.7 WR3 モードレジスタ3

モードレジスタ３は４軸各々が個別に持っています。どの軸のモードレジスタに書き込むかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したＮＯＰ命令を直前に書き込むことによって、書き込みたい軸を選択します。

モードレジスタ3は、マニュアル減速、加減速モード(対称／非対称、直線加減速／S字加減速)、ドライブパルス出力モード、エンコーダ入力モード、リミット信号端子交換、台形三角防止機能、タイマー繰り返しの設定を行うレジスタです。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR3	0	TMMD	AVTRI	LMINV	PIINV	PI-L	PIMD1	PIMD0	DPINV	DIR-L	DP-L	DPMD1	DPMD0	SACC	DSNDE	MANLD

D0 MANLD 加減速を行う定量パルスドライブにおける減速を自動減速にするか、マニュアル減速にするかを設定します。
0:自動減速、1:マニュアル減速
マニュアル減速モードにした場合は、マニュアル減速点(DP)が設定されていなければなりません。

D1 DSNDE 直線加減速ドライブの減速時の減速度を、加速度設定値にするか(対称)、個別の減速度設定値にするか(非対称)を設定します。また、S字加減速ドライブの減速時の減速度増加率を加速度増加率設定値にするか(対称)、個別の減速度増加率設定値にするか(非対称)を設定します。
0:対称加減速、1:非対称加減速
非対称S字加減速を行う定量パルスドライブでは、自動減速できませんので、D0(MANLD)ビットを1にし、マニュアル減速点(DP)を設定しなければなりません。

D2 SACC 加減速ドライブ時の速度カーブを直線加減速にするか、S字加減速にするかを設定します。
0:直線加減速、1:S字加減速
S字加減速の場合は、加速度増加率(JK)、(減速度増加率(DJ))が設定されていなければなりません。

D4,3 DPMD1,0 ドライブパルスの出力方式を設定します。

D4(DPMD1)	D3(DPMD0)	ドライブパルス出力方式
0	0	独立２パルス方式
0	1	１パルス・方向方式
1	0	２相パルス４逓倍方式
1	1	２相パルス２逓倍方式

独立2パルス方式にすると、出力信号 nPP に＋方向パルスが、出力信号 nPM に－方向パルスが出力されます。
1パルス・方向方式にすると、出力信号 nPLS に＋／－両方向のドライブパルスが、出力信号 nDIR にパルスの方向信号が出力されます。
2 相パルス方式にすると出力信号 nPA に 2 相パルスの A 相信号が、出力信号 nPB に 2 相パルスの B 相信号が出力されます。

D5 DP-L ドライブパルスの論理レベルを設定します。
0:正論理パルス、1:負論理パルス

正論理パルス： 負論理パルス：

D6 DIR-L ドライブパルス出力方式を1パルス・方向方式に設定したときの、ドライブパルスの方向出力信号の論理レベルを設定します。
このビットの値により、nDIR 出力信号の電圧レベルは下表のように出力されます。

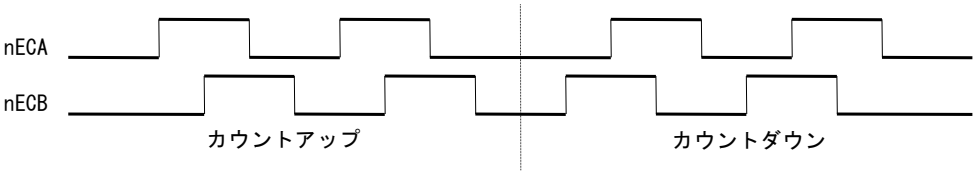
D6(DIR-L)	＋方向パルス出力時	－方向パルス出力時
0	Low	Hi
1	Hi	Low

D7 DPINV ドライブパルス出力の nPP/PLS/PA 信号と nPM/DIR/PB 信号の出力端子を入れ替えます。
0:初期状態、1:端子入れ替え
このビットを1にすると、ドライブパルス出力方式が独立2パルス方式のとき、＋方向のドライブでは、nPM 信号にドライブパルスが出力され、－方向のドライブでは nPP 信号にドライブパルスが出力されます。ほかのドライブパルス出力方式でも、同様に信号出力端子が入れ替わります。

D9,8 PIMD1,0 エンコーダ入力パルス方式を設定します。
エンコーダ入力信号は、実位置カウンタをカウントアップ／ダウンします。

D9(PIMD1)	D3(PIMD0)	エンコーダ入力パルス方式
0	0	２相パルス入力 ４逓倍
0	1	２相パルス入力 ２逓倍
1	0	２相パルス入力 １逓倍
1	1	アップ／ダウンパルス入力

このビットを2相パルス入力のモードに設定すると、正論理パルスでA相が進んでいるときはカウントアップ、B相が進んでいるときはカウントダウンします。4 逓倍設定時は両信号の↑、↓でカウントアップ、ダウンします。2 逓倍設定時は A 相信号の↑、↓でカウントアップ、ダウンします。1 逓倍設定時は B 相信号 Low 時の A 相信号の↑でカウントアップ、B 相信号 Low 時の A 相信号↓でカウントダウンします。



このビットをアップ／ダウンパルス入力のモードに設定すると、nPPIN 信号がカウントアップ入力に、nPMIN 信号がカウントダウン入力になります。それぞれ、正パルスの↑でカウントします。

D10 PI-L エンコーダ入力信号の正論理／負論理を設定します。
0:正論理、1:負論理
この設定により、エンコーダ入力パルス方式のアップ／ダウンパルス方式において、負パルスの↓でカウントになります。

D11 PIINV エンコーダ入力パルスの nECA/PPIN 信号と nECB/PMIN 信号の入力端子を入れ替えます。
0:初期状態、1:端子入れ替え
これにより、以下のように実位置カウンタの増減を反転させます。

D11(PIINV)	エンコーダ入力パルス方式	実位置カウンタ(RP)の増減
0	２相パルス入力	Ａ相が進んでいるときカウントアップする。 Ｂ相が進んでいるときカウントダウンする。
	アップ／ダウンパルス入力	nPPIN パルス入力のときカウントアップする。 nPMIN パルス入力のときカウントダウンする。
1	２相パルス入力	Ｂ相が進んでいるときカウントアップする。 Ａ相が進んでいるときカウントダウンする。
	アップ／ダウンパルス入力	nPMIN パルス入力のときカウントアップする。 nPPIN パルス入力のときカウントダウンする。

D12 LMINV ハードウェアリミット入力信号 nLMTP, nLMTM の入力端子を入れ替えます。
0:初期状態、1:端子入れ替え
このビットを1にすると、nLMTP 信号はー方向のリミット信号として動作し、nLMTM 信号は＋方向のリミット信号として動作します。

D13 AVTRI 直線加減速の定量パルスドライブにおける三角波形防止機能の有効／無効を設定します。リセット時、三角波形防止機能は有効になっています。
0:有効、1:無効

D14 TMMD タイマー動作の 1 回／繰り返しを設定します。
0:1回、1:繰り返し

リセット時には、D15～D0 は、すべて0にセットされます。D15 ビットには常に0をセットしてください。

6.8 WR4 アウトプットレジスタ1

X軸汎用入出力信号 XPIO7～0 およびY軸汎用入出力信号 YPIO7～0 を汎用出力として使用するときの、出力を設定するレジスタです。
各ビットに0をセットすると Low レベルが、1をセットすると Hi レベルが出力されます。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR4	YPIO7	YPIO6	YPIO5	YPIO4	YPIO3	YPIO2	YPIO1	YPIO0	XPIO7	XPIO6	XPIO5	XPIO4	XPIO3	XPIO2	XPIO1	XPIO0

リセット時には、D15～D0 は、すべて0にセットされます。

6.9 WR5 アウトプットレジスタ2

Z軸汎用入出力信号 ZPIO7～0 およびU軸汎用入出力信号 UPIO7～0 を出力として使用するときの、出力を設定するレジスタです。
各ビットに0をセットすると Low レベルが、1をセットすると Hi レベルが出力されます。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR5	UPIO7	UPIO6	UPIO5	UPIO4	UPIO3	UPIO2	UPIO1	UPIO0	ZPIO7	ZPIO6	ZPIO5	ZPIO4	ZPIO3	ZPIO2	ZPIO1	ZPIO0

リセット時には、D15～D0 は、すべて0にセットされます。

6.10 WR6, 7 ライトデータレジスタ1, 2

データ書き込み命令のデータをセットするレジスタです。WR6 レジスタにはライトデータ下位 16 ビット(WD15～WD0)、WR7 レジスタにはライトデータ上位 16 ビット(WD31～WD16)をセットします。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	WD15	WD14	WD13	WD12	WD11	WD10	WD9	WD8	WD7	WD6	WD5	WD4	WD3	WD2	WD1	WD0

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR7	WD31	WD30	WD29	WD28	WD27	WD26	WD25	WD24	WD23	WD22	WD21	WD20	WD19	WD18	WD17	WD16

データ書き込み命令は、まず、各々の命令で指定されているデータ長のデータをこれらのライトデータレジスタに書き込みます。
ライトデータレジスタ WR6,7(8ビットデータバスの場合は WR6L,WR6H,WR7L,WR7H)は、どれから先に書いてもかまいません。
その後、コマンドレジスタに命令コードを書き込むと、ライトデータレジスタの内容が、内部の各々のレジスタに取り込まれます。

書き込まれる数値データはすべてバイナリー(2進数)です。また、負の値は2の補数で扱います。

各々の命令のデータは、必ず指定されているデータ長で設定してください。

リセット時には、WR6,WR7 レジスタの内容は、不定です。

6.11 RRO 主ステータスレジスタ

主ステータスレジスタは、各軸のドライブ、エラー状態を表示します。また、連続補間次データ可、円弧補間の象限、連続補間のプリバッファスタックカウンタ(SC)を表示します。

RRO	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
	HSTC3	HSTC2	HSTC1	HSTC0	CNEXT	ZONE2	ZONE1	ZONE0	U-ERR	Z-ERR	Y-ERR	X-ERR	U-DRV	Z-DRV	Y-DRV	X-DRV

D3～0 n-DRV 各軸のドライブ状態を表します。このビットに1が立っているときは、その軸がドライブパルスを出力中であることを示しています。0のときはその軸がドライブを終了していることを示しています。また、自動原点出し実行時やヘリカル演算実行時には、実行している間このビットが1になります。

サーボモータ位置決め完了用入力信号 nINPOS を有効に設定しているときは、ドライブパルスを出力後、nINPOS 信号がアクティブになってから、このビットが0に戻ります。

D7～4 n-ERR 各軸のエラー発生状態をまとめて表示します。すなわち、各軸の RR2 レジスタのエラービット(D7～D0)のうち、どれか1つでも1が立つと、このビットが1になります。マルチチップ補間時、サブチップでエラーが発生するとメインチップの主軸のエラービットが1になります。

補間ドライブ以外のドライブ時(自動原点出しも含む)は、エラー・終了ステータスクリア命令(79h)、または次ドライブ開始で、このビットが0に戻ります。補間ドライブ時は、必ずエラー・終了ステータスクリア命令(79h)を発行してエラー解除をしてください。エラー解除を行わないと、その後の補間ドライブが正常に動作しません。

D10～8 ZONEm 円弧補間ドライブにおいて、現在ドライブ中の象限を示します。

D10	D9	D8	現在ドライブ中の象限
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

D11 CNEXT 連続補間次データ書き込み可能を表します。補間ドライブが開始されると、プリバッファスタックカウンタが1から7の間は1になり、次のセグメントの補間データ(パラメータデータおよび補間命令)を書き込むことが可能になります。

D15～12 HSTC3～0 連続補間のプリバッファスタックカウンタ(SC)の値を示します。

D15	D14	D13	D12	スタックカウンタ(SC)の値
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8

連続補間のドライブ中、SC=8 のときは、プリバッファのスタックが上限であることを表しています。SC が 7 以下のときは、次のセグメントの補間データ(パラメータデータおよび補間命令)を書き込むことができます。SC=0 は補間データをすべて出力し終え、連続補間のドライブが終了したことを表します。

6.12 RR1 ステータスレジスタ1

ステータスレジスタ1は4軸各々が個別に持っています。どの軸のステータスレジスタを読み出すかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したNOP命令を直前に書き込むことによって、読み出したい軸を選択します。

ステータスレジスタ1は割り込みが発生した要因を表示するレジスタです。割り込みが発生すると、その割り込み発生要因のビットが1になります。
割り込みを発生させるには、WR1 レジスタで、各要因ごとに割り込み許可に設定しておく必要があります。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR1	SYNC3	SYNC2	SYNC1	SYNC0	SPLTE	SPLTP	TIMER	H-END	D-END	C-END	C-STA	D-STA	CMR3	CMR2	CMR1	CMR0
	割り込み発生要因															

D3～0 CMR3～0 多目的レジスタMR3～0と比較対象との比較結果が、比較条件を満たすように変化して割り込みが発生したことを示します。
MR3～0 の比較対象および比較条件の設定は、多目的レジスタモード設定命令(20h)で行います。

D4 D-STA ドライブ開始で割り込みが発生したことを示します。

D5 C-STA 加減速ドライブ時に、定速域でのパルス出力の開始で、割り込みが発生したことを示します。

D6 C-END 加減速ドライブ時に、定速域でのパルス出力の終了で、割り込みが発生したことを示します。

D7 D-END ドライブ終了で割り込みが発生したことを示します。

D8 H-END 自動原点出し終了で割り込みが発生したことを示します。

D9 TIMER タイマーのタイムアップで割り込みが発生したことを示します。

D10 SPLTP スプリットパルスごとのパルスの↑で割り込みが発生したことを示します。
（スプリットパルス論理:Hi パルス設定時）

D11 SPLTE スプリットパルス終了で割り込みが発生したことを示します。

D15～12 SYNC3～0 同期動作 SYNC3～0 の起動で割り込みが発生したことを示します。

ある割り込み要因の割り込みが発生すると、このレジスタのビットが1になり、割り込み出力信号(INT0N)が Low レベルになります。CPUが、RR1 レジスタを読み出すと、RR1 レジスタのビットは0にクリアされ、割り込み出力信号(INT0N)はノンアクティブレベルに戻ります。

【注意】

- 8ビットデータバスの場合は、RR1L レジスタの読み出しで RR1L がクリアされ、RR1H の読み出しで RR1H がクリアされます。RR1L の読み出しで RR1H がクリアされることはありません。また、RR1H の読み出しで RR1L がクリアされることはありません。
- I^2C シリアルインターフェイスバスの場合は、RR1L レジスタと RR1H レジスタを分けて読み出すことはせず、必ず 2 バイト(RR1L,RR1H)を一度の通信で読み出すようにしてください。

6.13 RR2 ステータスレジスタ2

ステータスレジスタ2は4軸各々が個別に持っています。どの軸のステータスレジスタを読み出すかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したNOP命令を直前に書き込むことによって、読み出したい軸を選択します。

ステータスレジスタ2は、エラー情報、およびドライブ終了ステータスを表示するレジスタです。エラー情報(D7～D0)は、各ビットに1が立つと、ドライブ中にそのビットのエラーが発生したことを示します。この RR2 レジスタの D7～D0 のいずれかのビットに1が立つと、RR0 主ステータスレジスタの該当する軸の n-ERR ビットが1になります。

RR2 レジスタの各ビットが 1 になった場合は、エラー要因、またはドライブ終了要因が解消されても 1 が保持されます。補間ドライブ以外のドライブ時(自動原点出しも含む)のエラーは、エラー・終了ステータスクリア命令(79h)、または次のドライブ開始で全ビットが 0 に戻ります。補間ドライブ時のエラーは、必ずエラー・終了ステータスクリア命令(79h)を発行してエラービットを 0 に戻してください。

RR2	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
	EMG	ALARM	LMTM	LMTP	STOP2	STOP1	STOP0	SYNC	CERR	HOME	EMG	ALARM	HLMT-	HLMT+	SLMT-	SLMT+
	ドライブ終了ステータス								エラー情報							

D0 SLMT+ ソフトウェアリミット機能を有効にして、＋方向ドライブ中に、比較対象の位置カウンタが SLMT+値以上に大きくなったとき 1 になり、ドライブは停止します。

D1 SLMT- ソフトウェアリミット機能を有効にして、－方向ドライブ中に、比較対象の位置カウンタが SLMT-値より小さくなったとき 1 になり、ドライブは停止します。

D2 HLMT+ ハードウェアリミット信号を有効にして、＋方向ドライブ中に、nLMTP リミット信号がアクティブレベルになったとき 1 になり、ドライブは停止します。

D3 HLMT- ハードウェアリミット信号を有効にして、－方向ドライブ中に、nLMTM リミット信号がアクティブレベルになったとき 1 になり、ドライブは停止します。

D4 ALARM サーボモータアラーム用入力信号を有効にして、ドライブ中に nALARM 信号がアクティブレベルになったとき 1 になり、ドライブは停止します。

D5 EMG ドライブ中に、緊急停止信号 EMGN が Low レベルになったとき 1 になり、ドライブは停止します。

D6 HOME 自動原点出し実行時のエラーです。ステップ3開始時に、すでにエンコーダZ相信号 nSTOP2 がアクティブになっていると1が立ちます。

D7 CERR 補間に関するエラーで、連続補間のドライブ中に次のセグメントの補間データの書き込みが間に合わなくて停止、あるいはマルチチップ補間時の終点データ転送エラーのどちらかが発生したとき1になります。マルチチップ補間時のデータ転送エラーが発生したときは、RR3 レジスタのページ1の D12 ビットも1になります。

【注意】

- ドライブ中に進行方向のハード／ソフトリミットが作動すると、ドライブは減速停止または即停止します。停止後は停止要因が解消されるまでは、同方向へドライブ命令を発行しても再びエラーが発生し、ドライブ命令は実行されません。
- エラー情報ビットは、ドライブ停止中にそれぞれの要因がアクティブになっても1になりません。ソフトウェアリミット、ハードウェアリミットは、逆方向ドライブ時にそれぞれの要因がアクティブになってもエラーは発生しません。
- D7 ビットが1になった時は、必ずエラー・終了ステータスクリア命令(79h)を発行してください。エラーを解除しない場合、その後の補間ドライブが正常に動作しません。

D8 SYNC ドライブが、同期動作 SYNC3～0 のいずれかによって停止したとき、1になります。

D11～9 STOP2～0 ドライブが、外部停止信号 nSTOP2～0 によって停止したとき、1になります。

D12 LMT+ ドライブが、＋方向リミット信号 nLMTP によって停止したとき、1になります。

D13 LMT- ドライブが、－方向リミット信号 nLMTM によって停止したとき、1になります。

D14 ALARM ドライブが、サーボモータ用アラーム信号 nALARM によって停止したとき、1になります。

D15 EMG ドライブが、緊急停止信号 EMGN によって停止したとき、1になります。

ドライブ終了ステータス(D15～D8)は、ドライブを終了させた要因を示すビットです。ドライブを終了させた要因はドライブ終了ステータス(D15～D8)が示す要因以外にも、下記の 3 つの要因があります。

a. 定量パルスドライブにおいて、出力パルスをすべて出し終えたとき。
b. 減速停止、または即停止命令が書き込まれたとき。
c. ソフトウェアリミットが有効設定でアクティブになったとき。

ドライブ終了ステータス(D15～D8)の確認は、必ず RR0 主ステータスレジスタの n-DRV ビットでドライブが終了したことを確認した後に行ってください。

6.14 RR3 ステータスレジスタ3

ステータスレジスタ3は4軸各々が個別に持っています。どの軸のステータスレジスタを読み出すかは、直前に書き込んだ命令の軸指定によって決まります。あるいは、軸指定したNOP命令を直前に書き込むことによって、読み出したい軸を選択します。

ステータスレジスタ3は、ページ 0 およびページ 1 の 2 種類存在します。ページ 0 は、入力信号の状態を直接表示するレジスタです。また、自動原点出し実行時の実行ステートも表示します。ページ 1 は、同期動作の有効／無効設定、ドライブ中の加減速ドライブの加減速状態、S字加減速の加速度増加／減少状態、タイマー作動状態、スプリットパルス作動状態を表示します。また、マルチチップ補間時の転送エラー状態も表示します。
RR3 ページ表示命令(7Ah、7Bh)を書き込むことでページの指定をします。リセット時はページ 0 となります。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RR3 ページ0	0	HSST5	HSST4	HSST3	HSST2	HSST1	HSST0	LMTM	LMTP	ALARM	INPOS	ECB	ECA	STOP2	STOP1	STOP0
RR3 ページ1	1	0	0	MCERR	SPLIT	TIMER	ADSND	ACNST	AASND	DSND	CNST	ASND	SYNC3	SYNC2	SYNC1	SYNC0

■ ページ 0
各信号の入力信号状態ビットは、入力信号が Low レベルのときは0、Hi レベルのときは1を示します。D8～D0 の入力信号をファンクションとして使用しないときは、汎用入力信号として使用できます。以下の説明で、信号名の後の括弧内は X 軸から U 軸の順の端子番号を示しています。

D2～0 STOP2～0 外部停止信号 nSTOP2(70,91,110,129)、nSTOP1(73,92,111,130)、nSTOP0(74,93,112,131)の入力状態を表示します。

D3 ECA エンコーダ入力パルス信号 nECA/PPIN(45,47,49,51)の入力状態を表示します。
エンコーダ入力端子交換設定(WR3/D11:PIINV)をしても、本ビットが入力状態を表示する端子番号は変わりません。

D4 ECB エンコーダ入力パルス信号 nECB/PMIN(46,48,50,52)の入力状態を表示します。
エンコーダ入力端子交換設定(WR3/D11:PIINV)をしても、本ビットが入力状態を表示する端子番号は変わりません。

D5 INPOS サーボモータ位置決め完了用入力信号nINPOS(66,85,104,123)の入力状態を表示します。

D6 ALARM サーボモータアラーム用入力信号 nALARM(67,86,105,124)の入力状態を表示します。

D7 LMTP ハードウェアリミット入力信号 nLMTP(68,87,106,127)の入力状態を表示します。
ハードウェアリミット入力端子交換設定(WR3/D12:LMINV)をしても、本ビットが入力状態を表示する端子は変わりません。

D8 LMTM ハードウェアリミット入力信号 nLMTM(69,88,109,128)の入力状態を表示します。
ハードウェアリミット入力端子交換設定(WR3/D12:LMINV)をしても、本ビットが入力状態を表示する端子は変わりません。

D14～9 HSST5～0 自動原点出し実行ステートは、自動原点出し実行中に現在実行している動作内容を示します。
2.5.5 項を参照してください。

D15 PAGE RR3 がページ 0 を表示していることを示し、0 となります。

■ ページ 1

D3～0 SYNC3～0 同期動作 SYNC3～0 が有効状態のとき 1 になります。
同期動作を有効にするには、同期動作有効命令(8F～81h)を発行します。同期動作を無効にし、動作させないようにするには同期動作無効命令(9F～91h)を発行します。

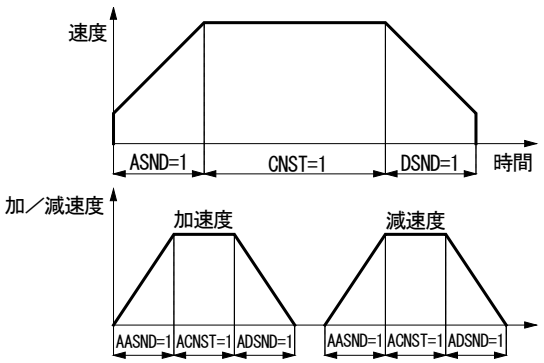
D4 ASND 加減速ドライブで、加速状態のとき1になります。

D5 CNST 加減速ドライブで、定速状態のとき1になります。

D6 DSND 加減速ドライブで、減速状態のとき1になります。

D7 AASND S字加減速ドライブで、加速度／減速度が増加状態のとき1になります。

D8 ACNST S字加減速ドライブで、加速度／減速度が一定状態のとき1になります。



D9 ADSND S字加減速ドライブで、加速度／減速度が減少状態のとき1になります。

D10 TIMER タイマー動作中に 1 になります。

D11 SPLIT スプリットパルス動作中に 1 になります。

D12 MCERR マルチチップ補間時の終点データ転送エラーが発生したとき、1になります。この場合、RR2 レジスタの D7 ビットも1になります。

D15 PAGE RR3 がページ 1 を表示していることを示し、1 となります。

6.15 RR4 PIOリードレジスタ1

X軸の汎用入出力信号 XPIO7～0、およびY軸の汎用入出力信号 YPIO7～0 の信号状態を表示するレジスタです。信号が Low レベルのときは0、Hi レベルのときは1を示します。

RR4

D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
YPI07	YPI06	YPI05	YPI04	YPI03	YPI02	YPI01	YPI00	XPI07	XPI06	XPI05	XPI04	XPI03	XPI02	XPI01	XPI00

D7～0 XPI07～0 X軸の汎用入出力信号 XPIO7～0 の状態を表示します。
XPIO7～0 信号を入力に設定しているときは入力状態を示し、出力に設定しているときは出力状態を示します。

D15～8 YPI07～0 Y 軸の汎用入出力信号 YPIO7～0 の状態を表示します。
YPIO7～0 信号を入力に設定しているときは入力状態を示し、出力に設定しているときは出力状態を示します。

6.16 RR5 PIOリードレジスタ2

Z軸の汎用入出力信号 ZPIO7～0、およびU軸の汎用入出力信号 UPIO7～0 の信号状態を表示するレジスタです。信号が Low レベルのときは0、Hi レベルのときは1を示します。

RR5

D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
UPI07	UPI06	UPI05	UPI04	UPI03	UPI02	UPI01	UPI00	ZPI07	ZPI06	ZPI05	ZPI04	ZPI03	ZPI02	ZPI01	ZPI00

D7～0 ZPI07～0 Z軸の汎用入出力信号 ZPIO7～0 の状態を表示します。
ZPIO7～0 信号を入力に設定しているときは入力状態を示し、出力に設定しているときは出力状態を示します。

D15～8 UPI07～0 U軸の汎用入出力信号 UPIO7～0 の状態を表示します。
UPIO7～0 信号を入力に設定しているときは入力状態を示し、出力に設定しているときは出力状態を示します。

6.17 RR6, 7 リードデータレジスタ1, 2

データ読み出し命令により、内部レジスタのデータがこれらのレジスタにセットされます。RR6 レジスタにはリードデータ下位 16 ビット(RD15～RD0)が、RR7 レジスタにはリードデータ上位 16 ビット(RD31～RD16)がセットされます。

RR6

D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0

RR7

D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
RD31	RD30	RD29	RD28	RD27	RD26	RD25	RD24	RD23	RD22	RD21	RD20	RD19	RD18	RD17	RD16

データはすべてバイナリー(2 進数)です。また、負の値は 2 の補数で扱います。

7. 命令

7.1 命令一覧

■ データ書き込み命令

コード	命　令	パラメータ記号	データ範囲	データ長（バイト）
００h	加速度増加率 設定	ＪＫ	1 ～ 1,073,741,823 [pps/sec^2]	4
０１	減速度増加率 設定	ＤＪ	1 ～ 1,073,741,823 [pps/sec^2]	4
０２	加速度 設定	ＡＣ	1 ～ 536,870,911 [pps/sec]	4
０３	減速度 設定	ＤＣ	1 ～ 536,870,911 [pps/sec]	4
０４	初速度 設定	ＳＶ	1 ～ 8,000,000 [pps]	4
０５	ドライブ速度 設定	ＤＶ	1 ～ 8,000,000 [pps]	4
０６	移動パルス数／終点 設定	ＴＰ	−2,147,483,646 ～ +2,147,483,646 (※1)	4
０７	マニュアル減速点 設定	ＤＰ	0 ～ 4,294,967,292	4
０８	円弧中心点 設定	ＣＴ	−1,073,741,823 ～ +1,073,741,823	4
０９	論理位置カウンタ 設定	ＬＰ	−2,147,483,648 ～ +2,147,483,647	4
０Ａ	実位置カウンタ 設定	ＲＰ	−2,147,483,648 ～ +2,147,483,647	4
０Ｂ	ソフトリミット＋ 設定	ＳＰ	−2,147,483,648 ～ +2,147,483,647	4
０Ｃ	ソフトリミット－ 設定	ＳＭ	−2,147,483,648 ～ +2,147,483,647	4
０Ｄ	加速カウンタオフセット 設定	ＡＯ	−32,768 ～ +32,767	2
０Ｅ	論理位置カウンタ最大値 設定	ＬＸ	1 ～ 2,147,483,647（7FFF FFFFh） または FFFF FFFFh	4
０Ｆ	実位置カウンタ最大値 設定	ＲＸ	1 ～ 2,147,483,647（7FFF FFFFh） または FFFF FFFFh	4
１０	多目的レジスタ０ 設定	ＭＲ０	−2,147,483,648 ～ +2,147,483,647	4
１１	多目的レジスタ１ 設定	ＭＲ１	−2,147,483,648 ～ +2,147,483,647	4
１２	多目的レジスタ２ 設定	ＭＲ２	−2,147,483,648 ～ +2,147,483,647	4
１３	多目的レジスタ３ 設定	ＭＲ３	−2,147,483,648 ～ +2,147,483,647	4
１４	原点検出速度 設定	ＨＶ	1 ～ 8,000,000 [pps]	4
１５	速度増減値 設定	ＩＶ	1 ～ 1,000,000 [pps]	4
１６	タイマー値 設定	ＴＭ	1 ～ 2,147,483,647 [μsec]	4
１７	スプリットパルス設定１	ＳＰ１	スプリット長：2 ～ 65,535 パルス幅：1 ～ 65,534	4
１８	スプリットパルス設定２	ＳＰ２	スプリットパルス数：0 ～ 65,535	2
１９	補間・終点最大値 設定	ＴＸ	0 ～ 1,073,741,823	4
１Ａ	ヘリカル回転数 設定	ＨＬＮ	0 ～ 65,535	2
１Ｂ	ヘリカル演算値 設定	ＨＬＶ	1 ～ 2,147,483,646	4

(※1)　但し、補間の終点データ範囲は、−1,073,741,823 ～ +1,073,741,823 になります。

【注意】

- データを書き込むときには必ず指定のデータ長で書き込んでください。
- 速度パラメータ値、タイマー値に記載の単位は、入力クロック(CLK)が 16MHz の場合です。入力クロック(CLK)が 16MHz 以外の場合のパラメータ計算式は、付録Bを参照してください。

■ モード書き込み命令

コード	命　令	パラメータ記号	データ長（バイト）
２０h	多目的レジスタモード設定	ＭＲＭ	2
２１	ＰＩＯ信号設定１	Ｐ１Ｍ	2
２２	ＰＩＯ信号設定２・その他設定	Ｐ２Ｍ	2
２３	自動原点出しモード設定１	Ｈ１Ｍ	2
２４	自動原点出しモード設定２	Ｈ２Ｍ	2
２５	入力信号フィルタモード設定	ＦＬＭ	2
２６	同期動作ＳＹＮＣ０設定	Ｓ０Ｍ	2
２７	同期動作ＳＹＮＣ１設定	Ｓ１Ｍ	2
２８	同期動作ＳＹＮＣ２設定	Ｓ２Ｍ	2
２９	同期動作ＳＹＮＣ３設定	Ｓ３Ｍ	2
２Ａ	補間モード設定	ＩＰＭ	2

【注意】データを書き込むときには必ず指定のデータ長で書き込んでください。

■ データ読み出し命令

コード	命　令	パラメータ記号	データ範囲	データ長（バイト）
３０h	論理位置カウンタ 読み出し	ＬＰ	-2,147,483,648 ～ +2,147,483,647	4
３１	実位置カウンタ 読み出し	ＲＰ	-2,147,483,648 ～ +2,147,483,647	4
３２	現在ドライブ速度 読み出し	ＣＶ	0 ～ 8,000,000 [pps]	4
３３	現在加減速度 読み出し	ＣＡ	0 ～ 536,870,911 [pps/sec]	4
３４	多目的レジスタ０ 読み出し	ＭＲ０	-2,147,483,648 ～ +2,147,483,647	4
３５	多目的レジスタ１ 読み出し	ＭＲ１	-2,147,483,648 ～ +2,147,483,647	4
３６	多目的レジスタ２ 読み出し	ＭＲ２	-2,147,483,648 ～ +2,147,483,647	4
３７	多目的レジスタ３ 読み出し	ＭＲ３	-2,147,483,648 ～ +2,147,483,647	4
３８	現在タイマー値 読み出し	ＣＴ	0 ～ 2,147,483,647 [μsec]	4
３９	補間・終点最大値	ＴＸ	0 ～ 1,073,741,823	4
３Ａ	現在ヘリカル回転数	ＣＨＬＮ	0 ～ 65,535	2
３Ｂ	ヘリカル演算値	ＨＬＶ	1 ～ 2,147,483,646	4
３Ｄ	ＷＲ１設定値 読み出し	ＷＲ１	（ビットデータ）	2
３Ｅ	ＷＲ２設定値 読み出し	ＷＲ２	（ビットデータ）	2
３Ｆ	ＷＲ３設定値 読み出し	ＷＲ３	（ビットデータ）	2
４０	多目的レジスタモード設定 読み出し	ＭＲＭ	（ビットデータ）	2
４１	ＰＩＯ信号設定１ 読み出し	Ｐ１Ｍ	（ビットデータ）	2
４２	ＰＩＯ信号設定２･その他設定 読み出し	Ｐ２Ｍ	（ビットデータ）	2
４３	加速度設定値 読み出し	ＡＣ	1 ～ 536,870,911 [pps/sec]	4
４４	初速度設定値 読み出し	ＳＶ	1 ～ 8,000,000 [pps]	4
４５	ドライブ速度設定値 読み出し	ＤＶ	1 ～ 8,000,000 [pps]	4
４６	移動パルス数／終点設定値 読み出し	ＴＰ	-2,147,483,646 ～ +2,147,483,646 (※1)	4
４７	スプリットパルス設定１ 読み出し	ＳＰ１	スプリット長：2 ～ 65,535 パルス幅：1 ～ 65,534	4
４８	汎用入力値 読み出し	ＵＩ	RR7:下位 1 バイト(PIN7～0) RR6:2 バイト(I2C 通信時の D15～0)	4

(※1)　但し、補間の終点データ範囲は、-1,073,741,823 ～ +1,073,741,823 になります。

■ ドライブ命令

コード	命　令
５０h	相対位置ドライブ
５１	反相対位置ドライブ
５２	＋方向連続パルスドライブ
５３	－方向連続パルスドライブ
５４	絶対位置ドライブ
５６	ドライブ減速停止
５７	ドライブ即停止
５８	方向信号＋設定
５９	方向信号－設定
５Ａ	自動原点出し実行

■ 補間命令

コード	命　令
６０h	１軸直線補間ドライブ(マルチチップ)
６１	２軸直線補間ドライブ
６２	３軸直線補間ドライブ
６３	４軸直線補間ドライブ
６４	ＣＷ円弧補間ドライブ
６５	ＣＣＷ円弧補間ドライブ
６６	２軸ビットパターン補間ドライブ
６７	３軸ビットパターン補間ドライブ
６８	４軸ビットパターン補間ドライブ
６９	ＣＷヘリカル補間ドライブ
６Ａ	ＣＣＷヘリカル補間ドライブ
６Ｂ	ＣＷヘリカル演算
６Ｃ	ＣＣＷヘリカル演算
６Ｄ	減速有効
６Ｅ	減速無効
６Ｆ	補間割り込みクリア／補間ステップ

■ 同期動作操作命令

コード	命　令
８１～８Ｆh	同期動作 有効設定
９１～９Ｆ	同期動作 無効設定
Ａ１～ＡＦ	同期動作 起動

■ その他命令

コード	命　令
７０h	速度増加
７１	速度減少
７２	偏差カウンタクリア出力
７３	タイマー始動
７４	タイマー停止
７５	スプリットパルス開始
７６	スプリットパルス停止
７７	ドライブ開始ホールド
７８	ドライブ開始フリー
７９	エラー・終了ステータスクリア
７Ａ	ＲＲ３ ページ０表示
７Ｂ	ＲＲ３ ページ１表示
７Ｃ	終点最大値クリア
１Ｆ	ＮＯＰ
００ＦＦ	コマンドリセット

【注意】これ以外の命令コードをコマンドレジスタに書き込まないでください。IC内部回路のテスト命令が起動し、思わぬ動作をする場合があります。

7.2 データ書き込み命令

データ書き込み命令は、書き込みデータを伴う命令です。ドライブのための、加速度、ドライブ速度、移動パルス数などの動作パラメータを設定します。複数の軸指定をすると、同じデータを指定した軸すべてに、同時にセットすることができます。

データ書き込み命令は、指定のデータ長が2バイトのときは WR6 レジスタに、データ長が4バイトのときは WR6,7 レジスタに数値をセットします。そののち、WR0 レジスタに軸指定と命令コードを書き込むと実行されます。

WR6,7 ライトデータレジスタにセットする数値データはすべてバイナリー(2進数)です。また、負の値は2の補数で扱います。

各々のデータは、必ず、データ範囲内の値を設定してください。範囲外の値を設定すると、正しい動作が行われません。

【注意】

- データ書き込み命令の命令処理に要する時間は、最大で 125nsec(CLK=16MHz の場合)です。命令を書き込んでからこの間は、次のデータや命令は書き込まないでください。
- 加速カウンタオフセット(AO)、論理位置カウンタ最大値(LX)、実位置カウンタ最大値(RX)を除く他のすべての動作パラメータは、リセット時は不定です。ドライブに必要なパラメータについては、ドライブ前に必ず適切な値を設定してください。
- 各速度パラメータ値、タイマー値に記載の単位は、入力クロック(CLK)が 16MHz の場合です。入力クロック(CLK)が 16MHz 以外の場合のパラメータ計算式は、付録Bを参照してください。

7.2.1 加速度増加率 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
００h	加速度増加率 設定	ＪＫ	1 ～ 1,073,741,823	4

加速度増加率設定値は、S字加減速における加速度の単位時間当たりの増加／減少率を決定するパラメータです。設定する値の単位は pps/sec^2 です。

加速度増加率 = JK [pps/sec^2]

加速と減速が対称なS字加減速ドライブ(WR3/D1=0)では、減速時にもこの加速度増加率の値が使用されます。

7.2.2 減速度増加率 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０１h	減速度増加率 設定	ＤＪ	1 ～ 1,073,741,823	4

減速度増加率設定値は、加速と減速が非対称なS字加減速ドライブ(WR3/D1= 1)における減速度の単位時間当たりの増加／減少率を決定するパラメータです。設定する値の単位は pps/sec^2 です。

減速度増加率 = DJ [pps/sec^2]

加速と減速が対称なS字加減速ドライブ(WR3/D1=0)では、減速度増加率の値は使用されません。

7.2.3 加速度 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０２h	加速度 設定	ＡＣ	1 ～ 536,870,911	4

直線加減速ドライブの加速時の加速度を決定するパラメータです。設定する値の単位は pps/sec です。

加速度 = AC [pps/sec]

加速と減速が対称な直線加減速ドライブ(WR3/D1=0)では、減速時にもこの加速度の値が使用されます。
S字加減速ドライブでは、このパラメータは最大値 536,870,911(1FFF FFFFh)をセットしてください。
部分S字加減速ドライブでは、このパラメータは直線加速部分の加速度をセットしてください。
加速と減速が対称な部分S字加減速ドライブ(WR3/D1=0)では、減速時にもこの加速度の値が使用されます。

ドライブ中の現在加速度値は、現在加減速度読み出し命令(33h)で読み出すことが可能です。
設定した加速度値は、加速度設定値読み出し命令(43h)で読み出すことが可能です。

7.2.4 減速度 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０３h	減速度 設定	ＤＣ	1 ～ 536,870,911	4

非対称な直線加減速ドライブ(WR3/D1=1)での、減速時の減速度となるパラメータです。設定する値の単位は pps/sec です。

減速度 = DC [pps/sec]

非対称なS字加減速ドライブでは、このパラメータは最大値 536,870,911(1FFF FFFFh)をセットしてください。
非対称な部分S字加減速ドライブでは、このパラメータは直線減速部分の減速度をセットしてください。

7.2.5 初速度 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０４h	初速度 設定	ＳＶ	1 ～ 8,000,000	4

加減速ドライブの加速開始の速度と減速終了時の速度です。設定する値の単位は pps です。

初速度 = SV [pps]

対象モータがステッピングモータの場合は、自起動周波数内の値を設定します。機械的共振周波数がある場合には、それを避けて初速度を設定します。

定量パルスドライブでは、初速度を極端に低い値に設定すると、尻切れ、引き摺りが発生する場合があります。
- 直線加減速ドライブの場合は加速度設定値の平方根の値以上を目安にしてください。
- S字加減速ドライブの場合には加速度増加率の平方根を 1/10 倍した値以上を目安にしてください。
- 部分S字加減速ドライブの場合は加速度設定値の平方根の値以上を目安にしてください。

直線加減速ドライブ $SV \geqq \sqrt{AC}$，**Ｓ字加減速ドライブ** $SV \geqq \sqrt{JK} \times 1/10$，**部分Ｓ字加減速ドライブ** $SV \geqq \sqrt{AC}$

設定した初速度値は、初速度設定値読み出し命令(44h)で読み出すことが可能です。

7.2.6 ドライブ速度 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０５h	ドライブ速度 設定	ＤＶ	1 ～ 8,000,000	4

加減速ドライブにおいて定速域に達したときの速度です。定速ドライブでは、始めからこの速度になります。設定する値の単位は pps です。

ドライブ速度 = DV [pps]

このドライブ速度を初速度以下に設定すると加減速ドライブは行われず、始めから定速ドライブになります。エンコーダのZ相サーチなど、低速でドライブし、信号を検出したら即停止させたい時は、ドライブ速度を初速度以下に設定します。

ドライブ速度は、ドライブ途中でも自由に変更することができます。加減速度ドライブの定速域でドライブ速度を再設定すると、再設定した速度に向かって加速または減速を始め、再設定した速度に達すると再び定速ドライブに移ります。

自動原点出しでは、このドライブ速度は、ステップ1の高速検出速度、および、ステップ4の高速移動速度になります。

【注意】
- S字加減速の定量パルスドライブ(自動減速モード時)、および非対称直線加減速の定量パルスドライブ(自動減速モード時)において、ドライブ途中でドライブ速度の変更はできません。
- S字加減速の連続パルスドライブは、定速域においてドライブ速度を変更することが可能ですが、加減速中のドライブ速度変更設定は無効になります。
- 対称直線加減速の定量パルスドライブにおいて、ドライブ途中にドライブ速度を変更することは可能ですが、変更頻度によって若干の尻切れ、引き摺りが発生する場合があります。
- 補間ドライブ中はドライブ速度の変更はできません。

ドライブ中の現在ドライブ速度値は、現在ドライブ速度読み出し命令(32h)で読み出すことが可能です。
設定したドライブ速度値は、ドライブ速度設定値読み出し命令(45h)で読み出すことが可能です。

7.2.7 移動パルス数／終点 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０６h	移動パルス数／終点 設定	ＴＰ	移動パルス数／絶対位置終点: -2,147,483,646 ～ +2,147,483,646 補間終点: -1,073,741,823 ～ +1,073,741,823	４

相対位置ドライブでは、現在位置からの移動パルス数を設定します。移動パルス数に正のパルス数を設定するとドライブ方向は＋方向になり、負のパルス数を設定するとドライブ方向は－方向になります。
反相対位置ドライブにおいて、移動パルス数に正のパルス数を指定すると、ドライブ方向は－方向になります。

絶対位置ドライブでは、原点(論理位置カウンタ=0)を基準にした移動先終点を符号付き 32 ビット値で設定します。

移動パルス数は、相対位置ドライブおよび反相対位置ドライブの途中で変更することができます。ただしドライブ方向が変わるような値に変更することはできません。また、すでに通過している位置に変更すると、ドライブが即停止しますのでご注意ください。
絶対位置ドライブの途中で、終点を変更することはできません。

直線、円弧補間ドライブでは、各軸の終点を設定します。終点座標は、31 ビットで現在位置に対する相対値を符号付きでセットします。

ビットパターン補間ドライブでは、各軸のビットデータを設定します。ビットデータは、32 ビットのうち下位 16 ビットに＋方向のビットデータを、上位 16 ビットに－方向のビットデータをセットします。

ヘリカル補間ドライブでは、X、Y 軸の終点および Z、U 軸の移動量を設定します。終点座標は、31 ビットで現在位置に対する相対値を符号付きでセットします。移動量は、回転数が 0 の場合は総移動量を、1以上の場合は1回転あたりの移動量を符号付きでセットします。

7.2.8 マニュアル減速点 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０７h	マニュアル減速点 設定	ＤＰ	0 ～ 4,294,967,292	４

マニュアル減速モード(WR3/D0=1)の加減速定量パルスドライブにおける、マニュアル減速点を設定します。マニュアル減速点は、定量パルスドライブの出力パルス数から、減速に消費するパルス数を引いた値を設定します。

マニュアル減速点 = 出力パルス数 - 減速消費パルス数

＜出力パルス数について＞
出力パルス数とは、定量パルスドライブにおいて実際に出力されるパルス数です。
相対位置ドライブにおいて、出力パルス数 P は移動パルス数 TP 設定値の絶対値です。
絶対位置ドライブにおいて、出力パルス数 P は移動パルス数 TP 設定値からドライブ開始前の論理位置カウンタ値 LP を減じた値の絶対値です。

相対位置ドライブ：出力パルス数 P = | TP |
絶対位置ドライブ：出力パルス数 P = | TP - LP |

7.2.9 円弧中心点 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０８h	円弧中心点 設定	ＣＴ	-1,073,741,823～+1,073,741,823	4

円弧補間、ヘリカル補間ドライブのときの中心点を設定します。中心座標は、現在位置に対する相対値を符号付きでセットします。

7.2.10 論理位置カウンタ 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０９h	論理位置カウンタ 設定	ＬＰ	-2,147,483,648 ～ +2,147,483,647	4

論理位置カウンタの値を設定します。

論理位置カウンタは、＋方向／－方向のドライブ出力パルスをアップ／ダウンカウントします。

論理位置カウンタの値は、常時書き込み可能です。論理位置カウンタ読出し命令(30h)で、常時読み出すこともできます。

7.2.11 実位置カウンタ 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０Ａh	実位置カウンタ 設定	ＲＰ	-2,147,483,648 ～ +2,147,483,647	4

実位置カウンタの値を設定します。

実位置カウンタは、エンコーダ入力パルスをアップ／ダウンカウントします。

実位置カウンタの値は、常時書き込み可能です。実位置カウンタ読出し命令(31h)で、常時読み出すこともできます。

7.2.12 ソフトリミット＋ 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０Ｂh	ソフトリミット＋ 設定	ＳＰ	-2,147,483,648 ～ +2,147,483,647	4

＋方向ソフトリミット SLMT+レジスタの値を設定します。
ソフトリミットの有効／無効、設定対象、停止モードの設定は、WR2 レジスタで行います。

ソフトリミット SLMT+レジスタの値は、常時書き込み可能です。

7.2.13 ソフトリミットー 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０Ch	ソフトリミットー 設定	ＳＭ	-2,147,483,648 ～ +2,147,483,647	4

ー方向ソフトリミット SLMT-レジスタの値を設定します。
ソフトリミットの有効／無効、設定対象、停止モードの設定は、WR2 レジスタで行います。

ソフトリミット SLMT-レジスタの値は、常時書き込み可能です。

7.2.14 加速カウンタオフセット 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０Dh	加速カウンタオフセット 設定	ＡＯ	-32,768 ～ +32,767	2

加速カウンタのオフセット値を設定します。
加速カウンタのオフセット値は、リセット時に 0 がセットされます。通常変更する必要はありません。

加速カウンタオフセットについては、2.1 節を参照してください。

本データ書き込み命令のデータ長は 2 バイトです。設定値は WR6 レジスタにのみ書き込みます。

7.2.15 論理位置カウンタ最大値 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０Eh	論理位置カウンタ最大値 設定	ＬＸ	1 ～ 2,147,483,647（7FFF FFFFh） またはFFFF FFFFh	4

論理位置カウンタの可変リング機能における、論理位置カウンタ最大値を正の値で設定します。

リセット時の値は FFFF FFFFh です。可変リング機能を使用しない場合は初期値のままにしておきます。

7.2.16 実位置カウンタ最大値 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
０Fh	実位置カウンタ最大値 設定	ＲＸ	1 ～ 2,147,483,647（7FFF FFFFh） またはFFFF FFFFh	4

実位置カウンタの可変リング機能における、実位置カウンタ最大値を正の値で設定します。

リセット時の値は FFFF FFFFh です。可変リング機能を使用しない場合は初期値のままにしておきます。

7.2.17 多目的レジスタ0 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１０h	多目的レジスタ０ 設定	ＭＲ０	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR0 の値を設定します。

多目的レジスタは、位置、速度、タイマーの値と大小比較や、同期動作として各種パラメータ値のロード／セーブなどに使用します。比較結果は比較出力信号の出力や、同期動作起動や割り込み発生に使用します。

多目的レジスタ MR0 の値は、常時書き込み可能です。多目的レジスタ0読出し命令(34h)で、常時読み出すこともできます。

7.2.18 多目的レジスタ1 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１１h	多目的レジスタ１ 設定	ＭＲ１	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR1 の値を設定します。

多目的レジスタは、位置、速度、タイマーの値と大小比較や、同期動作として各種パラメータ値のロード／セーブなどに使用します。比較結果は比較出力信号の出力や、同期動作起動や割り込み発生に使用します。

多目的レジスタ MR1 の値は、常時書き込み可能です。多目的レジスタ1読出し命令(35h)で、常時読み出すこともできます。

7.2.19 多目的レジスタ2 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１２h	多目的レジスタ２ 設定	ＭＲ２	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR2 の値を設定します。

多目的レジスタは、位置、速度、タイマーの値と大小比較や、同期動作として各種パラメータ値のロード／セーブなどに使用します。比較結果は比較出力信号の出力や、同期動作起動や割り込み発生に使用します。

多目的レジスタ MR2 の値は、常時書き込み可能です。多目的レジスタ2読出し命令(36h)で、常時読み出すこともできます。

7.2.20 多目的レジスタ3 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１３h	多目的レジスタ３ 設定	ＭＲ３	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR3 の値を設定します。

多目的レジスタは、位置、速度、タイマーの値と大小比較や、同期動作として各種パラメータ値のロード／セーブなどに使用します。比較結果は比較出力信号の出力や、同期動作起動や割り込み発生に使用します。

多目的レジスタ MR3 の値は、常時書き込み可能です。多目的レジスタ3読出し命令(37h)で、常時読み出すこともできます。

7.2.21 原点検出速度 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１４h	原点検出速度 設定	ＨＶ	1 ～ 8,000,000	4

自動原点出しのステップ2, 3の低速サーチ速度を設定します。設定する値の単位は pps です。

原点検出速度 = HV [pps]

検出信号がアクティブになったとき即停止させるために、初速度(SV)より低い値に設定します。

自動原点出しについては、2.5 節を参照してください。

7.2.22 速度増減値 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１５h	速度増減値 設定	ＩＶ	1 ～ 1,000,000	4

速度増加命令(70h)、速度減少命令(71h)によって、ドライブ中の現在速度を増減させる速度値を設定します。設定する値の単位は pps です。

速度増減値 = IV [pps]

加減速ドライブの定速域で速度増加／減少命令を発行すると、この速度増減値分だけ増加／減少した速度に向かって加速または減速を始め、その速度に達すると再び定速ドライブに移ります。

7.2.23 タイマー値 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１６h	タイマー値 設定	ＴＭ	1 ～ 2,147,483,647	4

タイマーがタイムアップする時間を設定します。設定する値の単位はμsec です。

タイマー値 = TM [μsec]

タイマー動作中の現在タイマー値は、現在タイマー値読み出し命令(38h)で読み出すことが可能です。

7.2.24 スプリットパルス設定１

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲		データ長(ﾊﾞｲﾄ)
１７h	スプリットパルス設定１	ＳＰ１	WR6	スプリット長：2 ～ 65,535	4
			WR7	パルス幅：1 ～ 65,534	

スプリットパルスのスプリット長とパルス幅を設定します。スプリット長、パルス幅はドライブパルス単位です。
書き込みデータは、WR6 にスプリット長をセットし、WR7 にパルス幅をセットします。

スプリット長、パルス幅は、スプリットパルス出力中に変更することができます。スプリット長、パルス幅を再設定すると、再設定した値で、スプリットパルス出力が継続します。

本データ書き込み命令はデータ長が 4 バイトですので、スプリット長とパルス幅のどちらか一方だけ値を変更する場合でも、必ずWR6 と WR7 の両レジスタに適正な値を設定してください。

設定したスプリットパルス設定1(SP1)の値は、スプリットパルス設定1読み出し命令(47h)で読み出すことが可能です。

7.2.25 スプリットパルス設定２

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１８h	スプリットパルス設定２	ＳＰ２	スプリットパルス数：0, 1 ～ 65,535	2

出力するスプリットパルス数を設定します。スプリットパルス数に0を設定すると、コマンドまたは同期動作によりスプリットパルス出力が停止されるまで、スプリットパルスを出力し続けます。

スプリットパルス数は、スプリットパルス出力中に変更することができます。

本データ書き込み命令のデータ長は 2 バイトです。設定値は WR6 レジスタにのみ書き込みます。

7.2.26 補間・終点最大値 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１９h	補間・終点最大値 設定	ＴＸ	1 ～ 1,073,741,823	4

直線補間時の終点の最大値を設定します。本命令は軸指定が不要です。終点最大値は符号無し 31 ビット値で設定します。
本命令で設定された値をもとに、直線補間の演算が行われます。

本命令を使用するときは、補間モード設定命令(2Ah)で直線補間最大値をマニュアル設定にします。

7.2.27 ヘリカル回転数 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１Ａh	ヘリカル回転数 設定	ＨＬＮ	0～65,535	2

ヘリカル補間時のヘリカル回転数を設定します。本命令は軸指定が不要です。

ヘリカル補間ドライブ中の現在ヘリカル回転数は、現在ヘリカル回転数読み出し命令(3Ah)で読み出すことが可能です。

7.2.28 ヘリカル演算値 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
１Ｂh	ヘリカル演算値 設定	ＨＬＶ	1 ～ 2,147,483,646	4

ヘリカル補間時のヘリカル演算値を設定します。本命令は軸指定が不要です。

ヘリカル補間に関しては、3.3 節を参照してください。

7.3 モード書き込み命令

モード書き込み命令は、書き込みデータを伴う命令です。多目的レジスタ、自動原点出し、同期動作、補間ドライブなどの動作モードを設定します。複数の軸指定をすると、同じデータを指定した軸すべてに同時にセットすることができます。ただし、補間モード設定は軸指定が不要です。

モード書き込み命令は、すべて書き込みデータ長が 2 バイトです。WR6 レジスタの各ビットに適正値を設定したのちに、WR0 レジスタに命令コードを書き込むと、WR6 レジスタの内容がIC内部の各モード設定レジスタにセットされます。

リセット時にはIC内部の各モード設定レジスタの全てのビットは0にクリアされています。

【注意】 データ書き込み命令の命令処理に要する時間は、最大で 125nsec(CLK=16MHz の場合)です。命令を書き込んでからこの間は、次のデータや命令は書き込まないでください。

7.3.1 多目的レジスタモード設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２０h	多目的レジスタモード設定	ＭＲＭ	２

多目的レジスタ MR3～0 の値を比較する対象、および比較条件を設定します。MR3～0 のそれぞれ個別に比較対象、比較条件を設定できます。比較結果は、同期動作起動要因、割り込み発生要因、比較信号出力などに使用できます。

WR6	D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
	M3C1	M3C0	M3T1	M3T0	M2C1	M2C0	M2T1	M2T0	M1C1	M1C0	M1T1	M1T0	M0C1	M0C0	M0T1	M0T0
	MR3 比較条件		MR3 比較対象		MR2 比較条件		MR2 比較対象		MR1 比較条件		MR1 比較対象		MR0 比較条件		MR0 比較対象	

D1,0 **M0T1,0** MR0 の比較対象を設定します。

D3,2 **M0C1,0** MR0 の比較条件を設定します。

D5,4 **M1T1,0** MR1 の比較対象を設定します。

D7,6 **M1C1,0** MR1 の比較条件を設定します。

D9,8 **M2T1,0** MR2 の比較対象を設定します。

D11,10 **M2C1,0** MR2 の比較条件を設定します。

D13,12 **M3T1,0** MR3 の比較対象を設定します。

D15,14 **M3C1,0** MR3 の比較条件を設定します。

(k:0～3)

MkT1 ビット	MkT0 ビット	MRm 比較対象
0	0	論理位置カウンタ (LP)
0	1	実位置カウンタ (RP)
1	0	現在速度値 (CV)
1	1	現在タイマー値 (CT)

(k:0～3)

MkC1 ビット	MkC0 ビット	MRm 比較条件
0	0	比較対象≧MRm
0	1	比較対象＞MRm
1	0	比較対象＝MRm
1	1	比較対象＜MRm

多目的レジスタモード設定で設定した比較条件(MnC1,0 ビット)にかかわらず、MR3～0 とそれぞれの比較対象の大小比較結果を RR4 レジスタにて確認できます。
多目的レジスタについて、詳細は 2.4 節を参照してください。

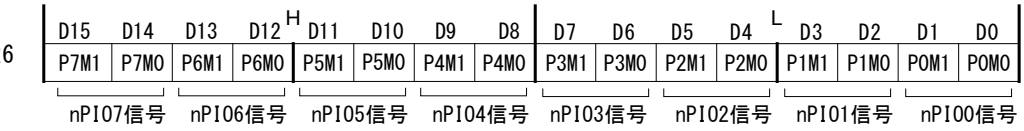
【注意】
比較対象を「現在速度値(CV)」、比較条件を「比較対象＝MRm」に設定しているとき、加減速ドライブで加速度，減速度が 4,194,304(400000h)pps/sec を超える場合には、比較結果がアクティブにならないことがあります。
比較対象が「現在速度(CV)」で加速度，減速度がこの値以上になるときには、比較条件として「比較対象＝MRm」は使用せず、「比較対象≧MRm」など他の条件を使用してください。

リセット時には、D15～D0 は、すべて0にセットされます。

7.3.2 PIO信号設定1

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２１h	ＰＩＯ信号設定１	Ｐ１Ｍ	２

nPIO7～0 信号の機能設定を行います。nPIO7～0 信号は、汎用入出力信号、同期入力信号、同期パルス出力信号、ドライブ状態出力信号、MRm 比較出力信号、外部信号によるドライブ操作のための入力信号に使用することができます。



D1,0　P0M1,0　nPIO0 信号の機能設定を行います。

D3,2　P1M1,0　nPIO1 信号の機能設定を行います。

D5,4　P2M1,0　nPIO2 信号の機能設定を行います。

D7,6　P3M1,0　nPIO3 信号の機能設定を行います。

D9,8　P4M1,0　nPIO4 信号の機能設定を行います。

D11,10　P5M1,0　nPIO5 信号の機能設定を行います。

D13,12　P6M1,0　nPIO6 信号の機能設定を行います。

D15,14　P7M1,0　nPIO7 信号の機能設定を行います。

設定する各機能を下表に示します。

(k:0～7)

PkM1 ビット	PkM0 ビット	機能
0	0	汎用入力 nPIO7～0 信号が入力状態になります。信号レベルは、X 軸は RR4 レジスタの D7～0、Y 軸は RR4 レジスタの D15～8 で読み出すことができます。同様に、Z 軸は RR5 レジスタの D7～0、U 軸は D15～8 で読み出すことができます。 同期動作では、信号の↑や↓で同期動作を起動することができます。 外部ドライブ操作では、nPIO4,5 信号によって相対位置ドライブや連続パルスドライブを起動することができます。
0	1	汎用出力 nPIO7～0 信号は出力状態になります。WR4 レジスタの D7～0 が X 軸の PIO7～0 に、D15～8 が Y 軸の PIO7～0 に出力されます。同様に、WR5 レジスタの D7～0 が Z 軸の PIO7～0 に、D15～8 が U 軸の PIO7～0 に出力されます。値が 0 で Low レベル、1 で Hi レベルが出力されます。
1	0	ドライブ状態出力 nPIO7～0 信号は出力状態になり、各信号は下表に示すドライブ状態を出力します。
1	1	同期パルス・MRm 比較出力 nPIO7～0 信号は出力状態になります。nPIO3～0 は同期パルスを、nPIO7～4 は MRm 比較値を出力します。比較対象と比較条件は、多目的レジスタモード設定命令(20h)で設定します

各 nPIOm 信号の機能を下表に示します。

(k:0～7)

nPIOm 信号 (端子番号)	PkM1,0 = 0,0	PkM1,0 = 0,1	PkM1,0 = 1,0	PkM1,0 = 1,1
	汎用入力 [*注]	汎用出力	ドライブ状態出力 (真で Hi)	同期パルス出力, MRm 比較出力
XPIO0(63) YPIO0(82) ZPIO0(101) UPIO0(120)	X 軸：RR4/D0 Y 軸：RR4/D8 Z 軸：RR5/D0 U 軸：RR5/D8 で信号レベル読み出し	X 軸：WR4/D0 Y 軸：WR4/D8 Z 軸：WR5/D0 U 軸：WR5/D8 値を出力	ドライブ中	SYNC0 同期パルス出力
XPIO1(62) YPIO1(81) ZPIO1(100) UPIO1(119)	X 軸：RR4/D1 Y 軸：RR4/D9 Z 軸：RR5/D1 U 軸：RR5/D9 で信号レベル読み出し	X 軸：WR4/D1 Y 軸：WR4/D9 Z 軸：WR5/D1 U 軸：WR5/D9 値を出力	エラー発生	SYNC1 同期パルス出力
XPIO2(61) YPIO2(80) ZPIO2(99) UPIO2(118)	X 軸：RR4/D2 Y 軸：RR4/D10 Z 軸：RR5/D2 U 軸：RR5/D10 で信号レベル読み出し	X 軸：WR4/D2 Y 軸：WR4/D10 Z 軸：WR5/D2 U 軸：WR5/D10 値を出力	加速中	SYNC2 同期パルス出力
XPIO3(60) YPIO3(79) ZPIO3(98) UPIO3(117)	X 軸：RR4/D3 Y 軸：RR4/D11 Z 軸：RR5/D3 U 軸：RR5/D11 で信号レベル読み出し	X 軸：WR4/D3 Y 軸：WR4/D11 Z 軸：WR5/D3 U 軸：WR5/D11 値を出力	定速中	SYNC3 同期パルス出力
XPIO4(59) YPIO4(78) ZPIO4(97) UPIO4(116)	X 軸：RR4/D4 Y 軸：RR4/D12 Z 軸：RR5/D4 U 軸：RR5/D12 で信号レベル読み出し	X 軸：WR4/D4 Y 軸：WR4/D12 Z 軸：WR5/D4 U 軸：WR5/D12 値を出力	減速中	MR0 比較出力 (真で Hi)
XPIO5(58) YPIO5(77) ZPIO5(96) UPIO5(115)	X 軸：RR4/D5 Y 軸：RR4/D13 Z 軸：RR5/D5 U 軸：RR5/D13 で信号レベル読み出し	X 軸：WR4/D5 Y 軸：WR4/D13 Z 軸：WR5/D5 U 軸：WR5/D13 値を出力	加速度増加中	MR1 比較出力 (真で Hi)
XPIO6(57) YPIO6(76) ZPIO6(95) UPIO6(114)	X 軸：RR4/D6 Y 軸：RR4/D14 Z 軸：RR5/D6 U 軸：RR5/D14 で信号レベル読み出し	X 軸：WR4/D6 Y 軸：WR4/D14 Z 軸：WR5/D6 U 軸：WR5/D14 値を出力	加速度一定中	MR2 比較出力 (真で Hi)
XPIO7(56) YPIO7(75) ZPIO7(94) UPIO7(113)	X 軸：RR4/D7 Y 軸：RR4/D15 Z 軸：RR5/D7 U 軸：RR5/D15 で信号レベル読み出し	X 軸：WR4/D7 Y 軸：WR4/D15 Z 軸：WR5/D7 U 軸：WR5/D15 値を出力	加速度減少中	MR3 比較出力 (真で Hi)

nPIO7～0 信号の詳しい使用方法については、2.8 節を参照してください。

***注** nPIO7～0 信号は汎用入力モード(PkM1,0 = 0,0)で、同期動作の起動要因として使用できます。詳細は 2.6 節を参照してください。
nPIO4,5 信号は汎用入力モード(PkM1,0 = 0,0)で、外部信号によるドライブ操作のための入力信号(nEXPP, nEXPM 入力)として使用できます。詳細は 2.12.1 項を参照してください。

リセット時には、D15～D0 は、すべて0にセットされます。

7.3.3 PIO信号設定2・その他設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２２h	ＰＩＯ信号設定２・その他設定	Ｐ２Ｍ	２

同期パルス出力の論理、パルス幅を設定します。ほかに、エラー発生時に同期動作無効にする設定、外部信号によるドライブ操作のモード設定、およびスプリットパルス出力の論理、開始パルス有無の設定を行います。

WR6

D15	D14	D13	D12	H D11	D10	D9	D8	D7	D6	D5	D4	L D3	D2	D1	D0
0	0	0	0	SPLBP	SPLL	EXOP1	EXOP0	ERRDE	PW2	PW1	PW0	P3L	P2L	P1L	P0L
				スプリット パルス		外部ドライブ 操作モード			同期パルス出力						

D3～0　PnL　nPIOm(m:3～0)を同期パルス出力信号として使用するときの、パルスの論理を設定します。
0:正論理パルス、1:負論理パルス

正論理パルス：　負論理パルス：

D6～4　PW2～0　同期パルス出力信号の出力パルス幅を設定します。

(CLK=16MHz 時)

D6～4 (PW2～0)	出力パルス幅
0	125ｎsec
1	312ｎsec
2	1μsec
3	4μsec
4	16μsec
5	64μsec
6	256μsec
7	1msec

出力パルス幅

(正論理パルスの場合)

D7　ERRDE　同期動作 SYNC3～0 の有効状態を、エラーが発生(RR0/D7～4:n-ERR = 1)したときに無効にするか否かを設定します。
0:エラー時に無効にしない、1:エラー時に無効にする
このビットを1に設定すると、RR0 レジスタの n-ERR ビットに 1 が立つと、直ちに同期動作 SYNC3～0 が全て無効設定になります。

同期動作を再び有効にする場合には、RR0 レジスタの n-ERR ビットが 1 になっていると、同期動作 SYNC3～0 を有効に設定することはできません。エラー・終了ステータスクリア命令(79h)などにより n-ERR ビットをクリアしたのち、同期動作有効設定を行ってください。
エラー発生状況、および同期動作 SYNC3～0 の有効／無効設定状況は RR3 レジスタ ページ 1 にて確認することができます。

D9,8　EXOP1,0　外部入力信号(nEXPP, nEXPM)によるドライブ操作モードを設定します。

D9(EXOP1)	D8(EXOP0)	外部信号によるドライブ操作モード
0	0	外部信号によるドライブ操作無効
0	1	連続パルスドライブモード
1	0	相対位置ドライブモード
1	1	手動パルサーモード

D10　SPLL　スプリットパルス出力のパルス論理を設定します。
0:正論理パルス、1:負論理パルス

正論理パルス： 　負論理パルス：

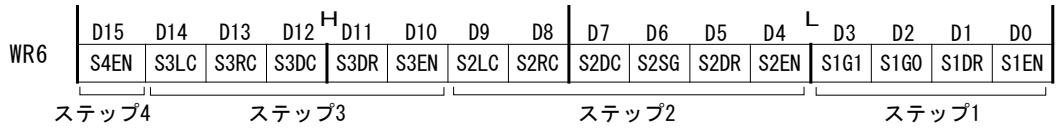
D11　SPLBP　スプリットパルス出力の開始パルス有無を設定します。
0:開始パルス無し、1:開始パルス有り

リセット時には、D15～D0 は、すべて0にセットされます。D15～D12 ビットには常に0をセットしてください。

7.3.4　自動原点出しモード設定1

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２３h	自動原点出しモード設定１	Ｈ１Ｍ	２

自動原点出しの動作モードを設定します。自動原点出し各ステップの有効／無効、検出方向、停止信号の選択、偏差カウンタクリア信号出力の有効／無効、位置カウンタクリアの設定を行います。



D0　S1EN　自動原点出しステップ1の動作「高速原点サーチ」を実行するか否かを設定します。
0:不実行、1:実行

D1　S1DR　ステップ1の検出方向を設定します。
0:＋方向、1:－方向

D3,2　S1G1,0　ステップ1の検出信号を設定します。
検出する入力信号の論理設定は WR2 レジスタで行います。

D3(S1G1)	D2(S1G0)	検出信号
0	0	nSTOP0
0	1	nSTOP1
1	0	リミット信号 *
1	1	（設定不可）

* リミット信号を指定すると、D1(S1DR)で指定した検出方向側のリミット信号が選択されます。

D4　S2EN　自動原点出しステップ2の動作「低速原点サーチ」を実行するか否かを設定します。
0:不実行、1:実行

D5　S2DR　ステップ2の検出方向を設定します。
0:＋方向、1:－方向

D6　S2SG　ステップ2の検出信号を設定します。
検出する入力信号の論理設定は WR2 レジスタで行います。

D6(S2SG)	検出信号
0	nSTOP1
1	リミット信号 *

* リミット信号を指定すると、D5(S2DR)で指定した検出方向側のリミット信号が選択されます。

D7 S2DC ステップ2の信号検出で、偏差カウンタクリア(nDCC)信号を出力するか否かを設定します。
0:出力しない、1:出力する

D8 S2RC ステップ2の信号検出で、実位置カウンタをクリアするか否かを設定します。
0:クリアしない、1:クリアする

D9 S2LC ステップ2の信号検出で、論理位置カウンタをクリアするか否かを設定します。
0:クリアしない、1:クリアする

D10 S3EN 自動原点出しステップ3の動作「低速Z相サーチ」を実行するか否かを設定します。
0:不実行、1:実行

D11 S3DR ステップ3の検出方向を設定します。
0:＋方向、1:－方向

D12 S3DC ステップ3の nSTOP2 信号で、偏差カウンタクリア(nDCC)信号を出力するか否かを設定します。
0:出力しない、1:出力する

D13 S3RC ステップ3の nSTOP2 信号検出で、実位置カウンタをクリアするか否かを設定します。
0:クリアしない、1:クリアする

D14 S3LC ステップ3の nSTOP2 信号検出で、論理位置カウンタをクリアするか否かを設定します。
0:クリアしない、1:クリアする

D15 S4EN ステップ4の動作「高速オフセット移動」を実行するか否かを設定します。
0:不実行、1:実行

自動原点出しの詳細については 2.5 節、および 2.5.4 項を参照してください。

リセット時には、D15～D0 は、すべて0にセットされます。

7.3.5 自動原点出しモード設定２

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２４h	自動原点出しモード設定２	Ｈ２Ｍ	２

自動原点出しの動作モードを設定します。自動原点出しステップ 3 の停止条件、位置カウンタクリア、偏差カウンタクリア出力、ステップ間タイマーの設定を行います。

	H D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4 L	D3	D2	D1	D0
WR6	0	0	0	0	0	HTM2	HTM1	HTM0	HTME	DCP2	DCP1	DCP0	DCPL	LCLR	RCLR	SAND
						ステップ間タイマー				偏差カウンタクリア出力						

D0 SAND 1に設定すると、ステップ3動作は、nSTOP1 信号がアクティブで、かつ、nSTOP2 信号がアクティブに変化したときに、停止します。
ステップ2の検出信号に nSTOP1 信号を選択したときのみ有効にできます。リミット信号を選択したときには有効にすることはできません。

D1 RCLR 自動原点出し終了時に実位置カウンタをクリアするか否かを設定します。
0:クリアしない、1:クリアする

D2 LCLR 自動原点出し終了時に論理位置カウンタをクリアするか否かを設定します。
0:クリアしない、1:クリアする

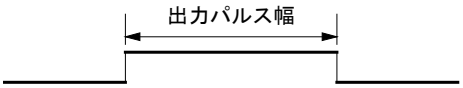
D3　DCPL　偏差カウンタクリア(nDCC)出力パルスの論理を設定します。
0:正論理パルス、1:負論理パルス

正論理パルス： 　負論理パルス：

D6～4　DCP2～0　偏差カウンタクリア(nDCC)出力のパルス幅を設定します。

(CLK=16MHz 時)

D6～4 (DCP2～0)	出力パルス幅
0	10μsec
1	20μsec
2	100μsec
3	200μsec
4	1msec
5	2msec
6	10msec
7	20msec



(正論理パルスの場合)

D7　HTME　ステップ間タイマーを有効にします。
0:無効、1:有効

D10～8　HTM2～0　ステップ間タイマーの時間幅を指定します。

(CLK=16MHz 時)

D10～8 (HTM2～0)	タイマー時間
0	1msec
1	2msec
2	10msec
3	20msec
4	100msec
5	200msec
6	500msec
7	1000msec

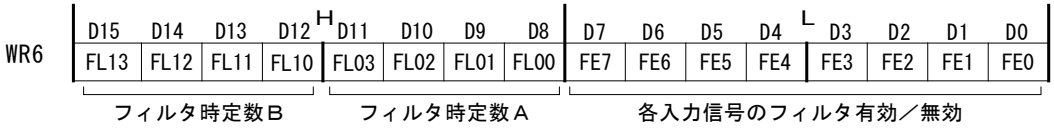
自動原点出しの詳細については 2.5 節、および 2.5.4 項を参照してください。

リセット時には、D15～D0 は、すべて0にセットされます。D15～D11 ビットには常に0をセットしてください。

7.3.6 入力信号フィルタモード 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２５h	入力信号フィルタモード設定２	ＦＬＭ	２

入力信号フィルタの有効／無効、および 2 つのフィルタの時定数を設定します。


WR6
H
L
D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0
FL13 FL12 FL11 FL10 FL03 FL02 FL01 FL00 FE7 FE6 FE5 FE4 FE3 FE2 FE1 FE0
フィルタ時定数Ｂ
フィルタ時定数Ａ
各入力信号のフィルタ有効／無効

D7～0　FE7～0　下表に示す入力信号ごとに、IC内蔵のフィルタ機能を有効にするか、無効(スルー)にするか設定します。
0:無効(スルー)、1:有効

指定ビット	入力信号	適用する時定数
D0(FE0)	EMGN	フィルタ時定数Ａ
D1(FE1)	nLMTP, nLMTM	
D2(FE2)	nSTOP0, nSTOP1	
D3(FE3)	nINPOS, nALARM	
D4(FE4)	nPI03～0	
D5(FE5)	nPI07～4	
D6(FE6)	nSTOP2	フィルタ時定数Ｂ
D7(FE7)	nECA, nECB	

D11～8　FL03～00　フィルタ時定数Aは D5～D0(FE5～0)で指定した入力信号フィルタの時定数を設定します。

D15～12 FL13～10　フィルタ時定数Bは D7, D6(FE7, 6)で指定した入力信号フィルタの時定数を設定します。

(CLK=16MHz 時)

時定数 (Hex)	除去可能な最大ノイズ幅	入力信号遅延時間
0	437.5ｎsec	500ｎsec
1	875ｎsec	1μsec
2	1.75μsec	2μsec
3	3.5μsec	4μsec
4	7μsec	8μsec
5	14μsec	16μsec
6	28μsec	32μsec
7	56μsec	64μsec
8	112μsec	128μsec
9	224μsec	256μsec
A	448μsec	512μsec
B	896μsec	1.024msec
C	1.792msec	2.048msec
D	3.584msec	4.096msec
E	7.168msec	8.192msec
F	14.336msec	16.384msec

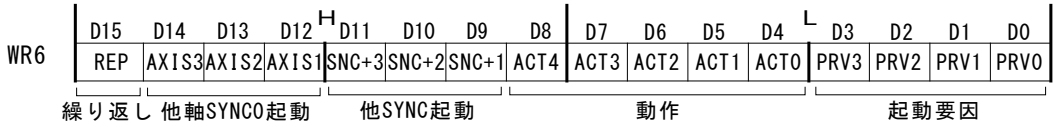
入力信号 EXPLSN、PIN7～0 に関しては、フィルタ機能を使用することはできません。
入力信号フィルタ機能の詳細は、2.11 節を参照してください。

リセット時には、D15～D0 は、すべて0にセットされます。

7.3.7 同期動作SYNC0，1，2，3設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２６h	同期動作ＳＹＮＣ０設定	ＳＯＭ	２
２７h	同期動作ＳＹＮＣ１設定	Ｓ１Ｍ	２
２８h	同期動作ＳＹＮＣ２設定	Ｓ２Ｍ	２
２９h	同期動作ＳＹＮＣ３設定	Ｓ３Ｍ	２

同期動作 SYNC0，1，2，3 の動作モードを設定します。各同期動作セットの起動要因、動作、他同期動作セット起動、他軸SYNC0 起動、同期動作の単一／繰り返しの設定を行います。



D3～0　PREV3～0　同期動作の起動要因をコード指定します。

(m：0,1,2,3)

コード (Hex)	SYNCm における起動要因	コード (Hex)	SYNCm における起動要因
0	NOP	8	スプリットパルス終了
1	MRm 比較が真に変化した	9	スプリットパルス出力
2	タイマーのタイムアップ	A	nPIOm 入力↑
3	ドライブ開始	B	nPIOm 入力↓
4	ドライブ定速域開始	C	nPIO(m+4)入力 Low かつ nPIOm 入力↑
5	ドライブ定速域終了	D	nPIO(m+4)入力 Hi かつ nPIOm 入力↑
6	ドライブ終了	E	nPIO(m+4)入力 Low かつ nPIOm 入力↓
7	スプリットパルス開始	F	nPIO(m+4)入力 Hi かつ nPIOm 入力↓

同期動作の起動要因、設定コードの詳細に関しては、2.6.1 項を参照してください。

D8～4　ACT4～0　同期動作の動作をコード指定します。

(m : 0,1,2,3)

コード (Hex)	SYNCm における動作	コード (Hex)	SYNCm における動作
00	NOP	0C	絶対位置ドライブ起動
01	ロード MRm → DV	0D	＋方向連続パルスドライブ起動
02	ロード MRm → TP	0E	－方向連続パルスドライブ起動
03	ロード MRm → SP1	0F	MRm 値の移動パルス数で相対位置ドライブ
04	ロード MRm → LP(SYNC0), RP(SYNC1), SV(SYNC2), AC(SYNC3)	10	MRm 値の終点へ絶対位置ドライブ
		11	ドライブ減速停止
05	セーブ LP → MRm	12	ドライブ即停止
06	セーブ RP → MRm	13	ドライブ速度増加
07	セーブ CT → MRm	14	ドライブ速度減少
08	セーブ CV(SYNC0), CA(SYNC1) → MRm	15	タイマー始動
09	同期パルス nPIOm 出力	16	タイマー停止
0A	相対位置ドライブ起動	17	スプリットパルス開始
0B	反相対位置ドライブ起動	18	スプリットパルス停止

DV : ドライブ速度　TP : 移動パルス数／終点　SP1 : スプリットパルス設定 1
LP : 論理位置カウンタ　RP : 実位置カウンタ　SV : 初速度
AC : 加速度　CT : 現在タイマー値　CV : 現在ドライブ速度
CA : 現在加減速度

同期動作の動作、設定コードの詳細に関しては、2.6.2 項を参照してください。

D11～9　SNC+3～1　同期動作の起動により、動作させるほかの同期動作セットを指定します。
0:無効、1:有効

自同期動作セット	D11(SNC+3)	D10(SNC+2)	D9(SNC+1)
SYNC0	SYNC3 起動	SYNC2 起動	SYNC1 起動
SYNC1	SYNC0 起動	SYNC3 起動	SYNC2 起動
SYNC2	SYNC1 起動	SYNC0 起動	SYNC3 起動
SYNC3	SYNC2 起動	SYNC1 起動	SYNC0 起動

D14～13 AXIS3～1　同期動作の起動により、動作させる他軸 SYNC0 を指定します。
0:無効、1:有効

自軸	D14(AXIS3)	D13(AXIS2)	D12(AXIS1)
X	U 軸 SYNC0 起動	Z 軸 SYNC0 起動	Y 軸 SYNC0 起動
Y	X 軸 SYNC0 起動	U 軸 SYNC0 起動	Z 軸 SYNC0 起動
Z	Y 軸 SYNC0 起動	X 軸 SYNC0 起動	U 軸 SYNC0 起動
U	Z 軸 SYNC0 起動	Y 軸 SYNC0 起動	X 軸 SYNC0 起動

D15　REP　同期動作セットの有効状態を、同期動作が一度起動したのち無効にするか否かを設定します。
0:無効にする(単一)、1:無効にしない(繰り返し)
このビットを 0 にすると、起動要因がアクティブになった初めの 1 回だけ同期動作が起動します。このビットを1にすると、起動要因がアクティブになるたびに繰り返し同期動作が起動します。

無効になった同期動作を再び有効にするには、同期動作有効命令を発行してください。同期動作 SYNC3～0 の有効／無効設定状況は RR0 レジスタにて確認することができます。

同期動作の詳細は、2.6 節を参照してください。

リセット時には、D15～D0 は、すべて0にセットされます。

7.3.8 補間モード 設定

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ長(ﾊﾞｲﾄ)
２Ah	補間モード設定	ＩＰＭ	２

補間ドライブ時のモードを設定します。本命令は軸指定が不要です。

	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WR6	INTB	INTA	0	MAXM	MLT1	MLT0	STEP	LMDF	SPD1	SPD0	0	CXIV	U-EN	Z-EN	Y-EN	X-EN

D3～0 U-EN～X-EN　補間ドライブを行う軸を指定します。下表にビットに対応した補間軸を示します。
0:補間軸として不使用、1:補間軸として使用

軸	指定ビット
X	D0
Y	D1
Z	D2
U	D3

主軸はX-EN＞Y-EN＞Z-EN＞U-EN の優先順で、1が設定されているビットの軸が選ばれます。

D4　CXIV　円弧補間を行うとき、補間軸を入れ替えて補間を行うかを指定します。
0:補間軸を入れ替えない、1:補間軸を入れ替える

D7,6　SPD1,0　補間ドライブの線速一定モードを設定します。

D7(SPD1)	D6(SPD0)	線速一定の種類
0	0	無効
0	1	２軸簡易
1	0	３軸簡易
1	1	２軸高精度

D8　LMDF　補間ドライブの短軸パルス均一化モードを設定します。
0:無効、1:有効

D9　STEP　補間ドライブの外部信号・命令ステップ送りを設定します。
0:無効、1:有効
このビットを1にすると、補間ドライブを外部信号(EXPLSN)あるいは補間ステップ命令(6Fh)でステップ送りにするモードになります。

D11,10　MLT1,0　マルチチップ補間を設定します。

D11(MLT1)	D10(MLT0)	マルチチップ補間の動作
0	0	無効
0	1	メインチップ
1	0	サブチップ
1	1	設定不可

マルチチップ補間を行う場合、メインチップは 01 を、それ以外(サブチップ)は 10 を設定します。
マルチチップ補間を行わない場合、00 を設定します。

D12 MAXM 直線補間最大値の設定方法を指定します。
0:自動設定、1:マニュアル設定
マニュアル設定にする場合、補間・終点最大値書き込み命令(19h)で終点最大値を設定します。

D15,14 INTB,A 連続補間ドライブ時の割り込みを設定します。
プリバッファのスタックカウンタの変化に応じて割り込みを発生させたい場合に設定します。

D15(INTB)	D14(INTA)	補間割り込みの発生
0	0	無効
0	1	スタックカウンタ４→３変化時
1	0	スタックカウンタ８→７変化時
1	1	スタックカウンタ８→７変化時 スタックカウンタ４→３変化時

割り込みが発生すると、補間割り込み出力信号(INT1N)が Low レベルになります。補間割り込みクリア命令発行、次のセグメントの補間実効命令発行、あるいは連続補間ドライブ終了のタイミングでクリアされ、補間割り込み出力信号はノンアクティブレベルに戻ります。

【注意】 補間ドライブを終了する場合は、WR6 レジスタに 0 を書き、本モード設定命令を発行して、補間モードをクリアするようにしてください。設定をしたまま通常ドライブを行うと、正しい動作にならない場合があります。

リセット時には、D15～D0 は、すべて0にセットされます。D5,13 ビットには常に0をセットしてください。

7.4 データ読み出し命令

データ読み出し命令は、各軸のレジスタの内容をリードデータレジスタに読み出す命令です。

WR0 レジスタにデータ読み出し命令コードを書き込むと、指定のデータが RR6,7 レジスタにセットされます。CPUは、RR6,7 レジスタを読み出すことによって指定のデータを得ることができます。
指定データは、データ長が2バイトのときは RR6 レジスタに、データ長が4バイトのときは RR6,7 レジスタにセットされます。

読み出しデータは、すべてバイナリー(2進数)です。また、負の値は2の補数で扱います。

【注意】
- データ読み出し命令の命令処理に要する時間は、最大で 125nsec(CLK=16MHz の場合)です。命令を書き込んでから、この時間ののち、RR6,7 レジスタを読み出してください。
- 各速度パラメータ値、タイマー値に記載の単位は、入力クロック(CLK)が 16MHz の場合です。入力クロック(CLK)が 16MHz 以外の場合のパラメータ計算式は、付録Bを参照してください。
- 軸指定は、必ず1軸のみの指定にしてください。

7.4.1 論理位置カウンタ 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３０h	論理位置カウンタ 読み出し	ＬＰ	-2,147,483,648 ～ +2,147,483,647	4

論理位置カウンタの現在値が、RR6,7 リードデータレジスタにセットされます。

7.4.2 実位置カウンタ 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３１h	実位置カウンタ 読み出し	ＲＰ	-2,147,483,648 ～ +2,147,483,647	4

実位置カウンタの現在値が、RR6,7 リードデータレジスタにセットされます。

7.4.3 現在ドライブ速度 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３２h	現在ドライブ速度 読み出し	ＣＶ	0 ～ 8,000,000	4

ドライブ中の現在ドライブ速度の値が、RR6,7 リードデータレジスタにセットされます。ドライブ停止時は0が読み出されます。
データの単位はドライブ速度設定値(DV)と同じ pps です。

補間ドライブにおいては、主軸の演算パルス速度のみが読み出し可能です。補間の主軸以外の軸は読み出すことはできません。

7.4.4 現在加減速度 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３３h	現在加減速度 読み出し	ＣＡ	0 ～ 536,870,911	4

加減速ドライブにおいて、加速中は現在加速度、減速中は現在減速度が、RR6,7 リードデータレジスタにセットされます。
ドライブ停止中は 0 が読み出されます。
データの単位は加速度設定値(AC)、減速度設定値(DC)と同じ pps/sec です。

【注意】
直線加減速ドライブ(対称)では、ドライブ中は常に設定加速度が読み出されます。
S 字加減速ドライブにおいて、定速域での現在加減速度読み出し値は無効です。

7.4.5 多目的レジスタ0 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３４h	多目的レジスタ０ 読み出し	ＭＲ０	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR0 の値が、RR6,7 リードデータレジスタにセットされます。
同期動作で MR0 にセーブされた現在位置や現在タイマー値、現在速度値を読み出すときに使用します。

7.4.6 多目的レジスタ1 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３５h	多目的レジスタ１ 読み出し	ＭＲ１	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR1 の値が、RR6,7 リードデータレジスタにセットされます。
同期動作で MR1 にセーブされた現在位置や現在タイマー値、現在加減速度値を読み出すときに使用します。

7.4.7 多目的レジスタ2 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３６h	多目的レジスタ２ 読み出し	ＭＲ２	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR2 の値が、RR6,7 リードデータレジスタにセットされます。
同期動作で MR2 にセーブされた現在位置や現在タイマー値を読み出すときに使用します。

7.4.8 多目的レジスタ3 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３７h	多目的レジスタ３ 読み出し	ＭＲ３	-2,147,483,648 ～ +2,147,483,647	4

多目的レジスタ MR3 の値が、RR6,7 リードデータレジスタにセットされます。
同期動作で MR3 にセーブされた現在位置や現在タイマー値を読み出すときに使用します。

7.4.9 現在タイマー値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３８h	現在タイマー値 読み出し	ＣＴ	0 ～ 2,147,483,647	4

タイマー動作中の現在タイマー値が、RR6,7 リードデータレジスタにセットされます。タイマー停止時は0が読み出されます。
データの単位はタイマー設定値(TM)と同じμ sec です。

7.4.10 補間・終点最大値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３９h	補間・終点最大値 読み出し	ＴＸ	1 ～ 1,073,741,823	4

直線補間の各軸の終点設定時点の終点最大値が RR6,7 リードデータレジスタにセットされます。本命令は軸指定が不要です。

補間ドライブの実行前と実行中では読み出される値が異なります。
補間ドライブ実行前は、入力中の補間セグメントの終点最大値が読み出されます。補間ドライブ実行中は現在実行中の補間セグメントの終点最大値が読み出されます。

7.4.11 現在ヘリカル回転数 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３Ah	現在ヘリカル回転数 読み出し	ＣＨＬＮ	0～65,535	2

ヘリカル補間動作中の現在ヘリカル回転数が、RR6 リードデータレジスタにセットされます。本命令は軸指定が不要です。

7.4.12 ヘリカル演算値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３Ｂh	ヘリカル演算値 読み出し	ＨＬＶ	1 ～ 2,147,483,646	４

ヘリカル演算命令(6Bh,6Ch)でヘリカル演算を行った結果を読み出すときに使用します。
演算結果が RR6,7 リードデータレジスタにセットされます。本命令は軸指定が不要です。

ヘリカル補間に関しては、3.3 節を参照してください。

7.4.13 WR1設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３Ｄh	ＷＲ１設定値 読み出し	ＷＲ１	（ビットデータ）	２

WR1 レジスタの設定値が、RR6 リードデータレジスタにセットされます。
WR1 レジスタアドレスにリードアクセスしても、WR1 レジスタに設定したデータは読み出せません。WR1 レジスタの設定値を確認したい場合、本命令にて読み出しを行ってください。

RR7 リードレジスタには 0 がセットされます。

7.4.14 WR2設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３Ｅh	ＷＲ２設定値 読み出し	ＷＲ２	（ビットデータ）	２

WR2 レジスタの設定値が、RR6 リードデータレジスタにセットされます。
WR2 レジスタアドレスにリードアクセスしても、WR2 レジスタに設定したデータは読み出せません。WR2 レジスタの設定値を確認したい場合、本命令にて読み出しを行ってください。

RR7 リードデータレジスタには 0 がセットされます。

7.4.15 WR3設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
３Ｆh	ＷＲ３設定値 読み出し	ＷＲ３	（ビットデータ）	２

WR3 レジスタの設定値が、RR6 リードデータレジスタにセットされます。
WR3 レジスタアドレスにリードアクセスしても、WR3 レジスタに設定したデータは読み出せません。WR3 レジスタの設定値を確認したい場合、本命令にて読み出しを行ってください。

RR7 リードデータレジスタには 0 がセットされます。

7.4.16 多目的レジスタモード設定 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４０h	多目的レジスタモード設定 読み出し	ＭＲＭ	（ビットデータ）	２

多目的レジスタモード設定命令(20h)で設定した値が、RR6 リードデータレジスタにセットされます。

RR7 リードデータレジスタには 0 がセットされます。

7.4.17 PIO信号設定1 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４１h	ＰＩＯ信号設定１ 読み出し	Ｐ１Ｍ	（ビットデータ）	２

PIO 信号設定1命令(21h)で設定した値が、RR6 リードデータレジスタにセットされます。

RR7 リードデータレジスタには 0 がセットされます。

7.4.18 PIO信号設定2 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４２h	ＰＩＯ信号設定２・その他設定 読み出し	Ｐ２Ｍ	（ビットデータ）	２

PIO信号設定2・その他設定命令(22h)で設定した値が、RR6 リードデータレジスタにセットされます。

RR7 リードレジスタには 0 がセットされます。

7.4.19 加速度設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４３h	加速度設定 読み出し	ＡＣ	1 ～ 536,870,911	4

加速度設定命令(02h)で設定した値が、RR6,7 リードデータレジスタにセットされます。
データの単位は pps/sec です。

同期動作によって加速度値(AC)へ MR3 値がロードされた場合には、その値が読み出されます。

7.4.20 初速度設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４４h	初速度設定値 読み出し	ＳＶ	1 ～ 8,000,000	4

初速度設定命令(04h)で設定した値が、RR6,7 リードデータレジスタにセットされます。
データの単位は pps です。

同期動作によって初速度値(SV)へ MR2 値がロードされた場合には、その値が読み出されます。

7.4.21 ドライブ速度設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４５h	ドライブ速度設定値 読み出し	ＤＶ	1 ～ 8,000,000	4

ドライブ速度設定命令(05h)で設定した値が、RR6,7 リードデータレジスタにセットされます。
データの単位は pps です。

同期動作によってドライブ速度値(DV)へ MRm 値がロードされた場合には、その値が読み出されます。

7.4.22 移動パルス数／終点設定値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４６h	移動パルス数／終点設定値 読み出し	ＴＰ	移動パルス数／絶対位置終点: -2,147,483,646 ～ +2,147,483,646 補間終点: -1,073,741,823 ～ +1,073,741,823	4

移動パルス数／終点速度設定命令(06h)で設定した値が、RR6,7 リードデータレジスタにセットされます。

同期動作によって移動パルス数／終点値(TP)へ MRm 値がロードされた場合には、その値が読み出されます。

7.4.23 スプリットパルス設定1 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲		データ長(ﾊﾞｲﾄ)
４７h	スプリットパルス設定１ 読み出し	ＳＰ１	RR6	スプリット長：2 ～ 65,535	4
			RR7	パルス幅：1 ～ 65,534	

スプリットパルス設定1命令(17h)で設定した値が、RR6,7 リードデータレジスタにセットされます。
RR6 レジスタにスプリット長が、RR7 レジスタにパルス幅がセットされます。

同期動作によってスプリットパルス設定1(SP1)へ MRm 値がロードされた場合には、その値が読み出されます。

7.4.24 汎用入力値 読み出し

命令コード	命　　令	ﾊﾟﾗﾒｰﾀ記号	データ範囲	データ長(ﾊﾞｲﾄ)
４８h	汎用入力値 読み出し	ＵＩ	RR7:下位１バイト(PIN7～0) RR6:2バイト(I2C通信時のD15～0)	4

本命令は軸指定が不要です。
I^2C シリアルインターフェイスバスモードの場合は、D15～0(端子番号 1～8、11～18)の信号レベルが RR6 リードレジスタにセットされます。I^2C シリアルインターフェイスバスモードでない場合は、RR6 リードレジスタは 0 となります。
PIN7～0(端子番号 132～139)を汎用入力として使用する場合、PIN7～0 の信号レベルが RR7 リードレジスタの下位 8 ビットにセットされます。上位 8 ビットは 0 となります。

	D15	D14	D13	D12 (H)	D11	D10	D9	D8	D7	D6	D5	D4 (L)	D3	D2	D1	D0
RR6	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

	D15	D14	D13	D12 (H)	D11	D10	D9	D8	D7	D6	D5	D4 (L)	D3	D2	D1	D0
RR7	0	0	0	0	0	0	0	0	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

信号が Low レベルのときは0、Hi レベルのときは1を示します。

7.5 ドライブ命令

ドライブ命令は、各軸のドライブパルスを出力する命令、およびそれに付随する命令です。

書き込みデータは伴わず、WR0 コマンドレジスタに軸指定と命令コードを書き込むと、直ちに実行されます。

ドライブ中は、RR0 主ステータスレジスタの n-DRV ビットに1が立ちます。ドライブが終了すると、n-DRV ビットは0に戻ります。

サーボモータドライバ用の nINPOS 信号を有効に設定しておくと、ドライブ終了後、nINPOS 入力信号がアクティブレベルになるのを待ってから、RR0 主ステータスレジスタの n-DRV ビットは0に戻ります。

【注意】ドライブ命令の命令処理に要する時間は、最大で 125nsec(CLK=16MHz の場合)です。次の命令を書き込むときは、この時間ののちに行ってください。

7.5.1 相対位置ドライブ

命令コード	命　　令
５０h	相対位置ドライブ

設定されている符号付き移動パルス数を＋方向ドライブパルス信号(nPP)、または－方向ドライブパルス信号(nPM)からパルス出力します。移動パルス数の値が正のときは nPP出力信号からパルス出力し、負のときは nPM出力信号からパルス出力します。(ドライブパルス出力方式:独立2パルス方式時)

ドライブ中は、＋方向のドライブパルスを1パルス出力するごとに論理位置カウンタが1つカウントアップし、－方向のドライブパルスを1パルス出力するごとに論理位置カウンタが1つカウントダウンします。

ドライブ命令を書き込む前に、出力させたい速度カーブに必要なパラメータと、移動パルス数が正しく設定されていなければなりません。

○：設定が必要

パラメータ	出力させたい速度カーブ				
	定速	対称直線加減速	非対称直線加減速	対称Ｓ字加減速	非対称Ｓ字加減速
加速度増加率（JK）				○	○
減速度増加率（DJ）					○
加速度（AC）		○	○	○ *	○ *
減速度（DC）			○		○ *
初速度（SV）	○	○	○	○	○
ドライブ速度（DV）	○	○	○	○	○
移動パルス数／終点（TP）	○	○	○	○	○
マニュアル減速点（DP）					○

*注：最大値 536,870,911（1FFF FFFFh）を設定します。ただし部分Ｓ字加減速では、直線加速／減速部分の加速度／減速度を設定します。

7.5.2 反相対位置ドライブ

命令コード	命　　令
５１h	反相対位置ドライブ

設定されている符号付き移動パルス数を＋方向ドライブパルス信号（nPP）、または－方向ドライブパルス信号（nPM）からパルス出力します。移動パルス数の値が正のときは nPM出力信号からパルス出力し、負のときは nPP出力信号からパルス出力します。（ドライブパルス出力方式：独立2パルス方式時）

本ドライブ命令は、定まった移動パルス数をドライブ命令によって方向を変えて出力する場合に使用します。
通常は移動パルス数（TP）に正のパルス量を設定しておき、＋方向に移動させたいときは相対位置ドライブ命令（50h）を、－方向に移動させたいときは反相対位置ドライブ命令（51h）を発行します。

ドライブ中は、＋方向のドライブパルスを1パルス出力するごとに論理位置カウンタが1つカウントアップし、－方向のドライブパルスを1パルス出力するごとに論理位置カウンタが1つカウントダウンします。

ドライブ命令を書き込む前に、出力させたい速度カーブに必要なパラメータと、移動パルス数が正しく設定されていなければなりません。

7.5.3 ＋方向連続パルスドライブ

命令コード	命　　令
５２h	＋方向連続パルスドライブ

停止コマンドまたは指定の外部信号がアクティブになるまで、連続して nPP出力信号からパルス出力します。（パルス出力方式：独立2パルス方式時）

ドライブ中は、ドライブパルスを1パルス出力するごとに論理位置カウンタが1つカウントアップします。

ドライブ命令を書き込む前に、出力させたい速度カーブに必要なパラメータが正しく設定されていなければなりません。

7.5.4 －方向連続パルスドライブ

命令コード	命　　令
５３h	－方向連続パルスドライブ

停止コマンドまたは指定の外部信号がアクティブになるまで、連続して nPM出力信号からパルス出力します。（パルス出力方式：独立2パルス方式時）

ドライブ中は、ドライブパルスを1パルス出力するごとに論理位置カウンタが1つカウントダウンします。

ドライブ命令を書き込む前に、出力させたい速度カーブに必要なパラメータが正しく設定されていなければなりません。

7.5.5 絶対位置ドライブ

命令コード	命　　令
５４h	絶対位置ドライブ

現在座標から終点座標までドライブを行います。
ドライブ前に、原点(論理位置カウンタ=0)を基準にした移動先終点を、移動パルス数／終点設定命令(06h)により符号付き 32 ビット値で設定します。

ドライブ命令を書き込む前に、出力させたい速度カーブに必要なパラメータと、終点が正しく設定されていなければなりません。

7.5.6 ドライブ減速停止

命令コード	命　　令
５６h	ドライブ減速停止

ドライブパルス出力を、途中で減速停止させます。ドライブ中の速度が初速度より低い場合には、即停止します。

補間ドライブ中においても、主軸に対して本命令またはドライブ即停止命令を書き込むと、補間ドライブは停止します。

ドライブが停止しているとき書き込んでも無処理となります。

7.5.7 ドライブ即停止

命令コード	命　　令
５７h	ドライブ即停止

ドライブパルス出力を、途中で即停止させます。加減速ドライブにおいても、即停止します。

ドライブが停止しているとき書き込んでも無処理となります。

7.5.8 方向信号＋設定

命令コード	命　　令
５８h	方向信号＋設定

ドライブパルス出力方式を 1 パルス・方向方式に設定したとき、ドライブ開始以前に方向信号DIRを＋方向のアクティブレベルにするための命令です。
11.2 節に示すように 1 パルス・方向方式でドライブを開始すると、方向信号が確定したのち 1CLK 後にドライブパルスの第 1 パルスが出力されます。ドライブパルスに対して方向信号のセットアップ時間を、その時間よりも長く取る必要がある場合に、本命令によって方向信号を＋方向に確定させます。

7.5.9 方向信号－設定

命令コード	命　　令
５９h	方向信号－設定

ドライブパルス出力方式を 1 パルス・方向方式に設定したとき、ドライブ開始以前に方向信号DIRを－方向のアクティブレベルにするための命令です。
11.2 節に示すように 1 パルス・方向方式でドライブを開始すると、方向信号が確定したのち 1CLK 後にドライブパルスの第 1 パルスが出力されます。ドライブパルスに対して方向信号のセットアップ時間を、その時間よりも長く取る必要がある場合に、本命令によって方向信号を－方向に確定させます。

7.5.10 自動原点出し実行

命令コード	命　　令
５Ah	自動原点出し実行

自動原点出しを実行します。
実行前に、自動原点出しモードや各パラメータを正しく設定しておく必要があります。自動原点出しの詳細は 2.5 節を参照してください。

7.6 補間命令

補間命令は、2軸／3軸／4軸直線補間、CW／CCW円弧補間、2軸／3軸／4軸ビットパターン補間、CW／CCWヘリカル補間および補間ドライブに付随する命令から成ります。補間命令は、WR0コマンドレジスタのD11～8ビットの軸指定の必要はありません。0をセットしてください。

いずれの補間を行う場合も、補間ドライブを開始する前に共通して必要なことは、次の2点です。

a. **補間を行う軸を指定する。（補間モード設定でセット。）**
b. **主軸に指定した軸の速度パラメータをセットする。**

補間ドライブ中は、RR0主ステータスレジスタの補間を行っている軸のn-DRVビットが1になり、ドライブが終了すると0に戻ります。

【注意】 補間命令の命令処理に要する時間は、最大で 125nSEC(CLK=16MHz の場合)です。次の命令を書き込むときは、この時間ののちに行ってください。

7.6.1 1軸直線補間ドライブ(マルチチップ)

命令コード	命　　令
６０h	１軸直線補間ドライブ(マルチチップ)

マルチチップ補間時のみ使用可能です。メインあるいはサブチップで補間軸を1軸のみ設定した時に使用します。

7.6.2 2軸直線補間ドライブ

命令コード	命　　令
６１h	２軸直線補間ドライブ

現在座標から終点座標まで2軸直線補間します。

ドライブ前に、補間を行う2軸のそれぞれの終点を相対値で出力パルス(P)にセットしておきます。

7.6.3 3軸直線補間ドライブ

命令コード	命　　令
６２h	３軸直線補間ドライブ

現在座標から終点座標まで3軸直線補間します。

ドライブ前に、補間を行う3軸のそれぞれの終点を相対値で出力パルス(P)にセットしておきます。

7.6.4 4軸直線補間ドライブ

命令コード	命　　令
６３h	４軸直線補間ドライブ

現在座標から終点座標まで4軸直線補間します。

ドライブ前に、補間を行う4軸のそれぞれの終点を相対値で出力パルス(P)にセットしておきます。

7.6.5 CW円弧補間ドライブ

命令コード	命　　令
６４h	ＣＷ円弧補間ドライブ

指定の中心座標を中心に、現在座標から終点座標まで時計方向に円弧補間します。

ドライブ前に、補間を行う２軸についてそれぞれ、現在位置に対する中心点を円弧中心点（Ｃ）に、現在位置に対する終点を出力パルス（Ｐ）に、相対値でセットしておきます。

終点座標を(0, 0)にセットすると、真円を描きます。

7.6.6 CCW円弧補間ドライブ

命令コード	命　　令
６５h	ＣＣＷ円弧補間ドライブ

指定の中心座標を中心に、現在座標から終点座標まで反時計方向に円弧補間します。

ドライブ前に、補間を行う２軸についてそれぞれ、現在位置に対する中心点を円弧中心点（Ｃ）に、現在位置に対する終点を出力パルス（Ｐ）に、相対値でセットしておきます。

終点座標を(0, 0)にセットすると、真円を描きます。

7.6.7 ２軸ビットパターン補間ドライブ

命令コード	命　　令
６６h	２軸ビットパターン補間ドライブ

2軸ビットパターン補間を行います。

ドライブ前に、補間を行う2軸の＋方向／－方向のビットデータをセットします。ドライブ前にセットできるビットデータは、各軸各方向とも16×8 = 128ビットまでです。これを越える場合はドライブ中に補充していきます。

7.6.8 3軸ビットパターン補間ドライブ

命令コード	命　　令
６７h	３軸ビットパターン補間ドライブ

3軸ビットパターン補間を行います。

ドライブ前に、補間を行う3軸の＋方向／－方向のビットデータをセットします。ドライブ前にセットできるビットデータは、各軸各方向とも16×8 = 128ビットまでです。これを越える場合はドライブ中に補充していきます。

7.6.9 4軸ビットパターン補間ドライブ

命令コード	命　　令
６８h	４軸ビットパターン補間ドライブ

4軸ビットパターン補間を行います。

ドライブ前に、補間を行う4軸の＋方向／－方向のビットデータをセットします。ドライブ前にセットできるビットデータは、各軸各方向とも16×8 = 128ビットまでです。これを越える場合はドライブ中に補充していきます。

7.6.10 CWヘリカル補間ドライブ

命令コード	命　　令
６９h	ＣＷヘリカル補間ドライブ

時計方向にヘリカル補間を行います。

ヘリカル補間に関しては、3.3 節を参照してください。

7.6.11 CCWヘリカル補間ドライブ

命令コード	命　　令
６Ａh	ＣＣＷヘリカル補間ドライブ

反時計方向にヘリカル補間を行います。

7.6.12 CWヘリカル演算

命令コード	命　　令
６Bh	CWヘリカル演算

時計方向へのヘリカル演算を行います。

本ICのヘリカル補間は、ZあるいはU軸の移動を均一に行なうために円弧補間の総出力パルス数を予め知っておく必要があります。ヘリカル演算命令はこの総出力パルス数を求めるための命令です。

ヘリカル補間に関しては、3.3 節を参照してください。

7.6.13 CCWヘリカル演算

命令コード	命　　令
６Ch	ＣＣWヘリカル演算

反時計方向のヘリカル演算を行います。

7.6.14 減速有効

命令コード	命　　令
６Dh	減速有効

加減速で補間ドライブを行うときの自動減速またはマニュアル減速を有効状態にします。

単独の補間ドライブを加減速で行うときには、ドライブ前に必ず本命令を発行する必要があります。連続補間では、減速させる補間セグメントの補間命令書き込みの前で、減速有効命令を書き込みます。

リセット時には、減速無効状態になります。本命令によって減速を有効状態にすると、補間ドライブが終了するか、減速無効命令(6Eh)が書き込まれるか、あるいはリセットするまで有効状態になります。

減速有効／無効は、補間ドライブのときだけ働きます。各軸を独自にドライブするときには、自動減速またはマニュアル減速は常に有効状態です。

7.6.15 減速無効

命令コード	命　　令
６Ｅh	減速無効

加減速で補間ドライブを行うときの自動減速またはマニュアル減速を無効状態にします。

7.6.16 補間割り込みクリア／補間ステップ

命令コード	命　　令
６Ｆh	補間割り込みクリア／補間ステップ

補間割り込みクリアは、連続補間で発生した割り込み(INT1N信号)をクリアします。

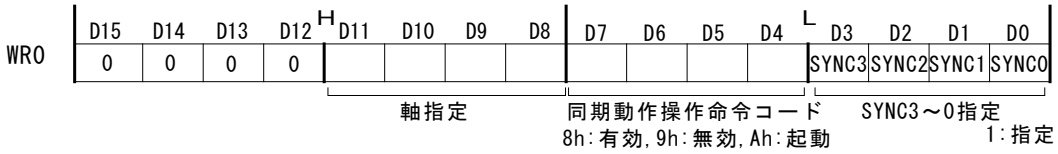
補間ステップは、補間ドライブを1パルスごとにステップ送りします。

割り込みに関しては2.10節を、補間ステップ送りに関しては3.9節を参照してください。

7.7 同期動作操作命令

同期動作操作命令は、同期動作を有効設定にする、無効設定にする、あるいは起動するための命令です。
4 つの同期動作セット SYNC3～0 のうち、任意の同期動作セットを同時に有効設定、無効設定、起動することができます。

同期動作操作命令は、WR0 コマンドレジスタの D7～D4 の 4 ビットに操作命令コードを、D3～D0 の 4 ビットに操作したい同期動作セットを指定します。すなわち D7～D4 には、同期動作を有効にしたいときには 8h、無効にしたいときには 9h、起動させたいときには Ah を設定します。また、D3～D0 は 4 つの同期動作セット SYNC3、SYNC2、SYNC1、SYNC0 に対応しており、操作したい同期動作セットに対応するビットを 1 にします。



書き込みデータは伴わず、WR0 コマンドレジスタに軸指定と命令コードを書き込むと実行されます。

【注意】 同期動作操作命令の命令処理に要する時間は、最大で 125nsec（CLK=16MHz の場合）です。次の命令を書き込むときは、この時間ののちに行ってください。

7.7.1 同期動作 有効設定

命令コード	命　　令
８１h ～８Ｆh	同期動作 有効設定

命令コードの下位 4 ビットでビット指定した各同期動作セットを有効に設定します。
同期動作有効設定を行う前に、有効に設定する同期動作セットのモード設定を同期動作 SYNC3～0 設定命令（29h～26h）にて行ってください。

■ 設定例：X 軸の同期動作セット SYNC0 と SYNC2 を有効にしたいときには、WR0 に 0185h を書き込みます。

同期動作 SYNC3～0 の有効／無効状態は RR3 レジスタ ページ 1 にて確認することができます。
リセット時は、SYNC3～0 は全て無効設定です。

【注意】
PIO 信号設定2・その他設定命令（22h）で、エラー発生で同期動作を無効とする設定（D7：ERRDE ビット=1）にして、かつエラー発生状態（RR0 レジスタの n-ERR ビットが 1）のとき、本命令を発行しても同期動作を有効に設定することはできません。エラー・終了ステータスクリア命令（79h）などにより n-ERR ビットをクリアしたのち、同期動作有効設定命令を発行してください。

7.7.2 同期動作 無効設定

命令コード	命　　令
９１h ～９Fh	同期動作 無効設定

命令コードの下位 4 ビットでビット指定した各同期動作セットを無効に設定します。
無効に設定した同期動作セットは、起動要因発生や同期動作起動命令によっても起動しません。

■ 設定例:X 軸の同期動作セット SYNC1 と SYNC3 を無効にしたいときには、WR0 に 019Ah を書き込みます

同期動作 SYNC3～0 の有効／無効設定状況は RR3 レジスタ ページ 1 にて確認することができます。
リセット時は、SYNC3～0 は全て無効設定です。

7.7.3 同期動作 起動

命令コード	命　　令
Ａ１h ～ＡFh	同期動作 起動

命令コードの下位 4 ビットでビット指定した各同期動作セットを、命令によって起動します。
同期動作起動前に、起動する同期動作セットのモード設定を、同期動作 SYNC3～0 設定命令(29h～26h)にて行う必要があります。また、同期動作有効設定命令によって、起動する同期動作セットを有効に設定しておく必要があります。
同期動作 SYNC3～0 の有効／無効設定状況は RR3 レジスタ ページ 1 にて確認することができます。

■ 設定例:X 軸の同期動作セット SYNC0 を起動するときには、WR0 に 01A1h を書き込みます
X 軸の同期動作セット SYNC3～0 すべてを起動するときには、WR0 に 01AFh を書き込みます。

7.8 その他の命令

書き込みデータは伴わず、軸指定とWR0 コマンドレジスタに命令コードを書き込むと、実行されます。

【注意】命令処理に要する時間は、最大で 125nsec（CLK=16MHz の場合）です。次の命令を書き込むときは、この時間ののちに行ってください。

7.8.1 速度増加

命令コード	命　　令
７０h	速度増加

ドライブ中に速度を速度増減値分だけ増加させます。

速度増減値（IV）は、速度増減値設定命令（15h）により事前に設定されていなければなりません。

本命令は、連続パルスドライブ中に使用します。定量パルスドライブ中には使用しないでください。定量パルスドライブ中に本命令を多用すると、ドライブ終了時に尻切れ・引き摺りが発生する場合があります。
また、S字加減速ドライブにおいては、加速中および減速中に本命令を発行しても無効となります。必ず定速中（RR3 ページ 1／D5：CNST＝1）に行ってください。

本命令の発行により、ドライブ速度設定値（DV）が更新されることはありません。

補間ドライブでは本命令は使用することはできません。

7.8.2 速度減少

命令コード	命　　令
７１h	速度減少

ドライブ中に速度を速度増減値分だけ減少させます。

速度増減値（IV）は、速度増減値設定命令（15h）により事前に設定されていなければなりません。

本命令は、連続パルスドライブ中に使用します。定量パルスドライブ中には使用しないでください。定量パルスドライブ中に本命令を多用すると、ドライブ終了時に尻切れ・引き摺りが発生する場合があります。
また、S字加減速ドライブにおいては、加速中および減速中に本命令を発行しても無効となります。必ず定速中（RR3 ページ 1／D5：CNST＝1）に行ってください。

本命令の発行により、ドライブ速度設定値（DV）が更新されることはありません。

補間ドライブでは本命令は使用することはできません。

7.8.3 偏差カウンタクリア出力

命令コード	命　　令
７２h	偏差カウンタクリア出力

nDCC 出力端子から偏差カウンタクリアパルスを出力します。
この命令を発行する前に、自動原点出しモード設定2命令（24h）で、パルスの論理レベル、パルス幅を設定する必要があります。
詳細は 2.5.2 項、2.5.4 項を参照してください。

7.8.4 タイマー始動

命令コード	命　　令
７３h	タイマー始動

タイマーを始動します。
本命令によりタイマーを始動すると現在タイマー値（CT）は 0 からカウントアップを開始し、タイマー値（TM）に指定された値になるとタイムアップします。
タイムアップ後に、タイマーを繰り返し動作させることも可能です。繰り返し動作を行うには WR3 レジスタの D14 ビット(TMMD)を1に設定します。

タイマー動作の詳細は、2.9 節を参照してください。

7.8.5 タイマー停止

命令コード	命　　令
７４h	タイマー停止

タイマーを停止します。
タイマーを途中で停止するとカウントアップしていた現在タイマー値（CT）は 0 に戻ります。次にタイマーを始動させると、0 から始まります。

7.8.6 スプリットパルス開始

命令コード	命　　令
７５h	スプリットパルス開始

スプリットパルスの出力を開始します。
スプリットパルスはドライブ中に nSPLTP 出力端子から出力されます。
スプリットパルス開始命令発行で、スプリットパルス動作中を示す RR3 レジスタ ページ1の SPLIT ビットが 1 になります。
本命令を発行する前に、スプリットパルス長など各パラメータを正しく設定する必要があります。

スプリットパルスの各パラメータの詳細は 2.7 節を参照してください。

7.8.7 スプリットパルス停止

命令コード	命　　令
７６h	スプリットパルス停止

スプリットパルスの出力を停止します。
スプリットパルス停止命令が発行されると、スプリットパルス動作中を示す RR3 レジスタ ページ 1 の SPLIT ビットが 0 になります。

スプリットパルス停止命令が発行されたとき、スプリットパルス出力信号がHiレベルに有るときには指定のパルス幅のHiレベルを確保してから終了します。(正論理設定時)

7.8.8 ドライブ開始ホールド

命令コード	命　　令
７７h	ドライブ開始ホールド

ドライブの開始を一時、停止します。

複数の軸のドライブを同時スタートさせるときに使用します。同時スタートさせたい軸に本命令を発行してから、それぞれの軸にドライブ命令を書き込みます。その後、それらの軸に、同時にドライブ開始フリー命令(78h)を書き込むと、全軸同時にドライブを開始します。

連続補間ドライブでは、ドライブ開始前に必要なセグメント分の補間データをプリバッファにセットする時に本命令を使用します。
連続補間については、3.7 節を参照してください。

7.8.9 ドライブ開始フリー

命令コード	命　　令
７８h	ドライブ開始フリー

ドライブ開始ホールド命令(77h)によってドライブ開始がホールドされている状態を解除します。

7.8.10 エラー・終了ステータスクリア

命令コード	命　　令
７９h	エラー・終了ステータスクリア

RR2 レジスタのすべてのエラー情報ビット、すべてのドライブ終了ステータスビット、および RR0 レジスタのエラービット(D7～4: n-ERR)を 0 にクリアします。

補間ドライブでエラーが発生したときのエラークリアも、本命令を使用します。

7.8.11 RR3 ページ0表示

命令コード	命　　令
７Ah	ＲＲ３ ページ０表示

RR3 レジスタの表示内容をページ 0 にします。
ページ 0 表示時は、RR3 レジスタの D15 ビットが 0 になります。

7.8.12 RR3 ページ1表示

命令コード	命　　令
７Bh	ＲＲ３ ページ１表示

RR3 レジスタの表示内容をページ1にします。
ページ1表示時は、RR3 レジスタの D15 ビットが1になります。

7.8.13 終点最大値クリア

命令コード	命　　令
７Ch	終点最大値クリア

直線補間において、現在書き込み中の補間終点に対して自動算出された終点最大値をクリアします。
本命令は軸指定が不要です。

7.8.14 NOP

命令コード	命　　令
１Fh	ＮＯＰ

命令は何も実行されません。

7.8.15 コマンドリセット

命令コード	命　　令
００ＦＦh	コマンドリセット

本 IC をリセットします。

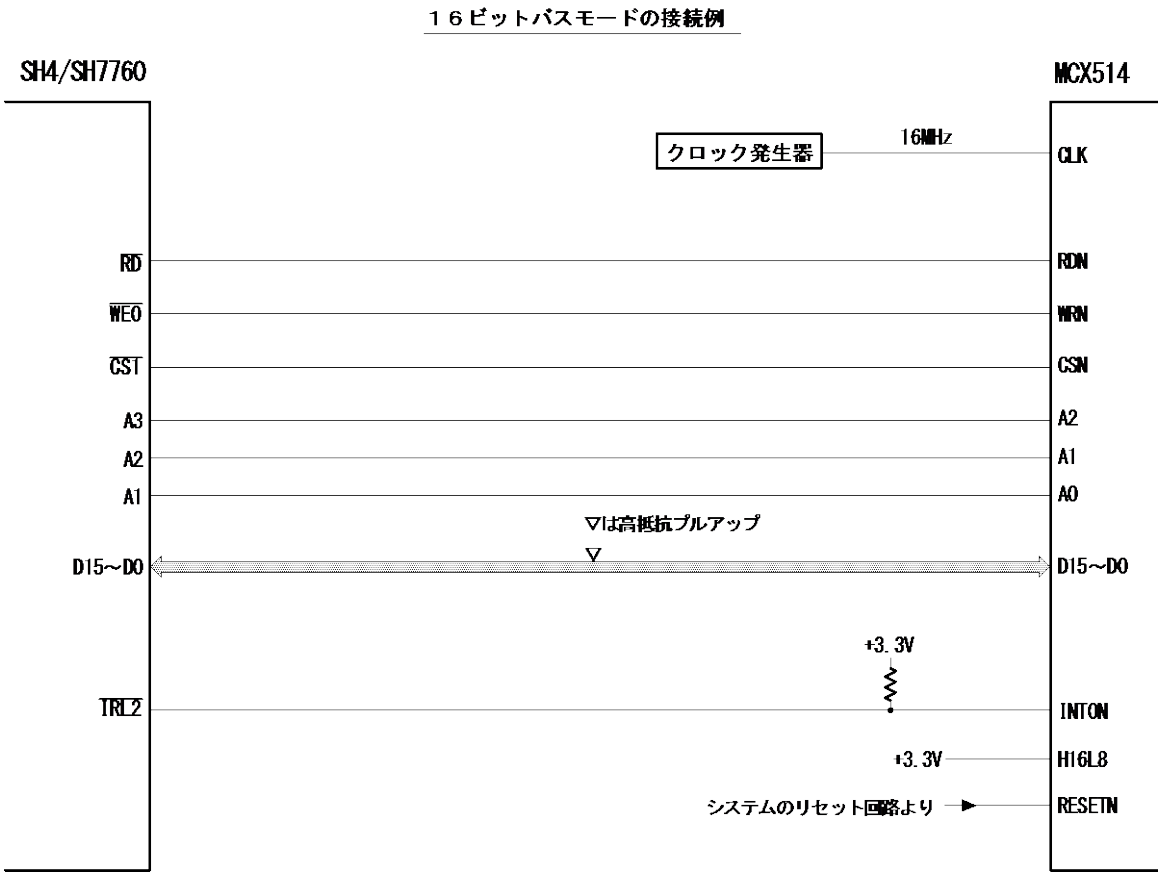
本命令は、WR0 レジスタの上位 8 ビット（D15～D8）を必ずすべて 0 に設定してください。
命令書き込み後 8CLK（500nsec：CLK=16MHz 時）の間、本 IC へのアクセスはできません。

本命令は、8ビットデータバス、I^2C シリアルインターフェイスバスにおいても、上位バイト(WR0H)の書き込みを行ってください。
上位バイト(WR0H)に 00h を必ず先に書き込み、下位バイト(WR0L)に FFh を後から書き込みます。下位バイトを書き込むと、直ちにリセットが実行されます。

8. 入出力信号接続例

8.1 16 ビット／8 ビットバスモードの接続例

■SH-4CPU との 16 ビットバスモード接続例

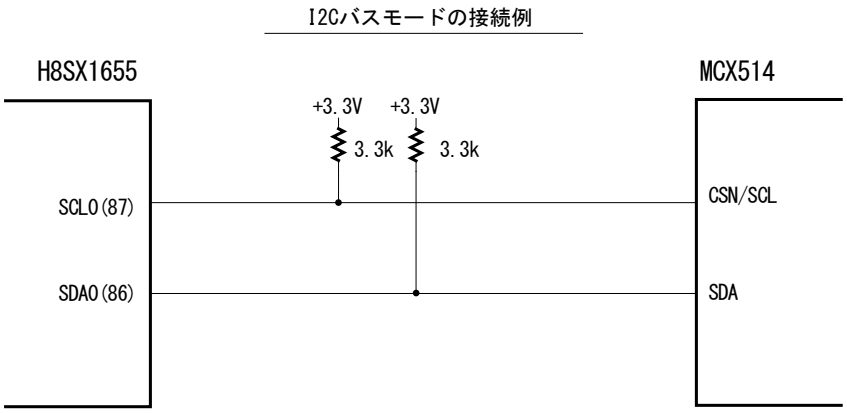


SH-4/SH7760 ウェイト制御例

バスクロック	66.664MHz	―
セットアップウェイト	１サイクル挿入	レジスタ設定：WCR3/A1S0=1
アクセスウェイト	２サイクル挿入	レジスタ設定：WCR2/A1W2, A1W1, A1W0 = 010
ホールドウェイト	１サイクル挿入	レジスタ設定：WCR3/A1H1, A1H0 = 01

8.2 I²C バスモードの接続例

■H8SX1655CPU との I²C バスモード接続例



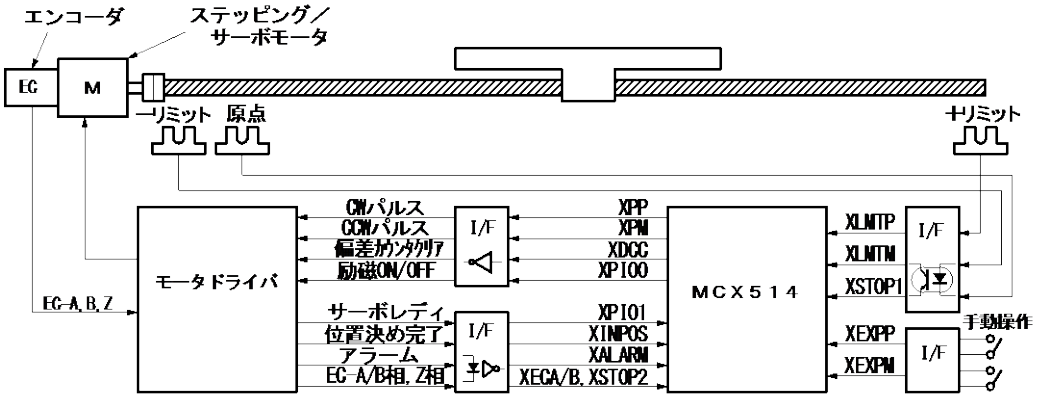
H8SX1655 レジスタ設定例

レジスタ	アドレス	設定値：8ビット(D7～D0)
ICCRA_0	H'FFEB0	10101001 (*)

(*) D7:I2C バスインターフェースイネーブル。1 で転送動作可能状態。
D5:マスタ／スレーブ選択。1 でマスタを指定。
D4:送信／受信選択。マスタ受信モードでは1、マスタ送信モードでは 0 とする(設定例は 0)。
D3～D0:転送クロックの選択。設定値では 200kbps(CLK＝16M 時)となる。

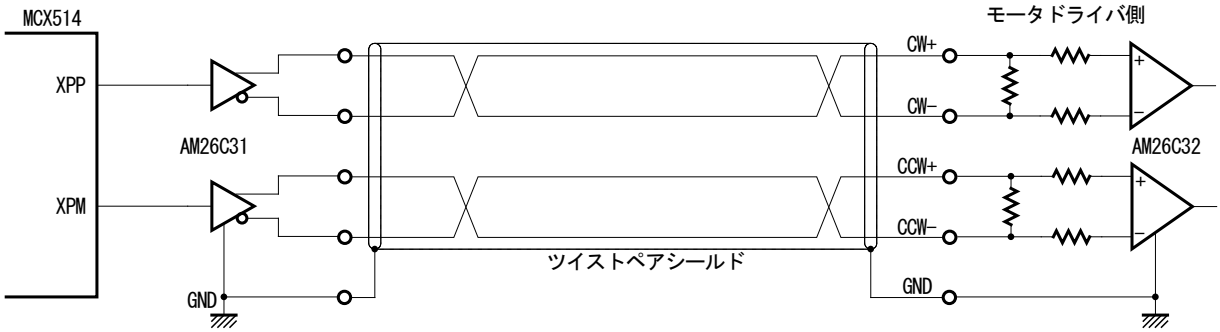
8.3 モーションシステム構成例

下の図は、モーションシステムのX軸分の例を示しています。4軸すべてについて、同様に構成を取ることができます。

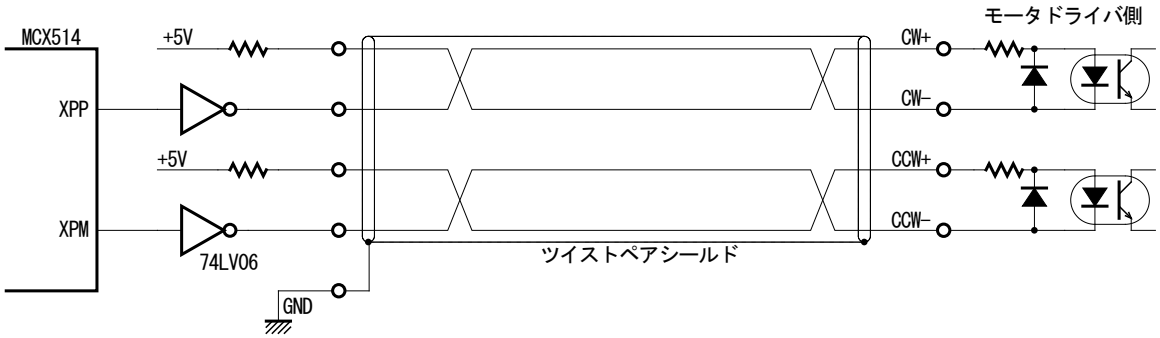


8.4 ドライブパルス出力回路例

■ 差動ラインドライバ出力



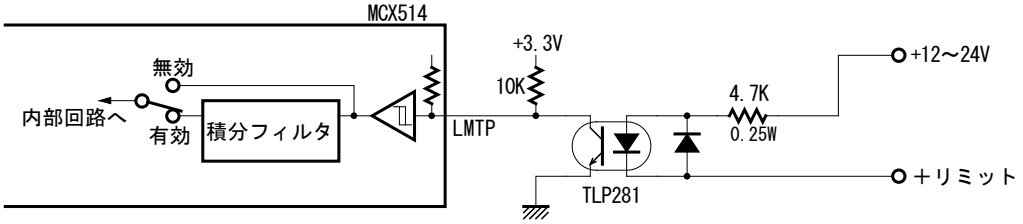
■ オープンコレクタ TTL 出力



ドライブパルス出力信号は、EMCを考慮して、ツイストペアシールド線を使用することをおすすめします。

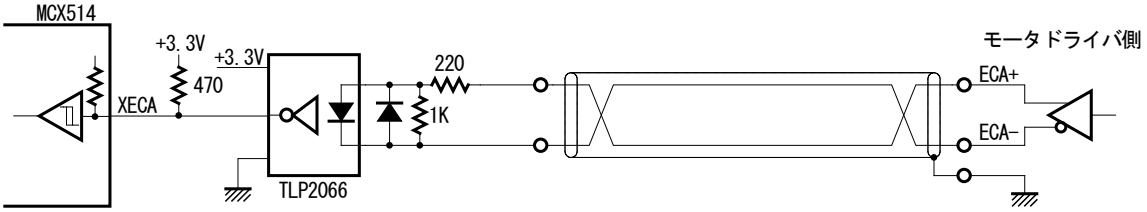
8.5 リミット等の入力信号の接続例

リミット信号等は、通常、配線をかなり引き回す場合が多く、ノイズも乗りやすくなります。フォトカプラだけではノイズを吸収できないことがあります。IC内のフィルタ機能を有効にして、適当な時定数(FL=Ah,Bh)を設定してください。



8.6 エンコーダ入力信号の接続例

下の図は、差動ラインドライバ出力のエンコーダ信号を高速フォトカプラICで受けて、MCX514 に入力する回路例です。



9. 制御プログラム例

この章では、C言語によるMCX514の制御プログラム例を示します。16ビットバス構成のプログラムです。
このプログラムは、弊社ホームページ（http://www.novaelec.co.jp/）からダウンロードできます。ファイル名：MCX514Apl.c

```
////////////////////////////////////////////////////////////////////////////
// 命令コードの定義
////////////////////////////////////////////////////////////////////////////
//////////////////////////////
// データ書き込み命令
//////////////////////////////
#define		MCX514_CMD00_JK		0x0000		// 加速度増加率 設定
#define		MCX514_CMD01_DJ		0x0001		// 減速度増加率 設定
#define		MCX514_CMD02_AC		0x0002		// 加速度 設定
#define		MCX514_CMD03_DC		0x0003		// 減速度 設定
#define		MCX514_CMD04_SV		0x0004		// 初速度 設定
#define		MCX514_CMD05_DV		0x0005		// ドライブ速度 設定
#define		MCX514_CMD06_TP		0x0006		// 移動パルス数/終点 設定
#define		MCX514_CMD07_DP		0x0007		// マニュアル減速点 設定
#define		MCX514_CMD09_LP		0x0009		// 論理位置カウンタ 設定
#define		MCX514_CMD0A_RP		0x000A		// 実位置カウンタ 設定
#define		MCX514_CMD0B_SP		0x000B		// ソフトリミット＋ 設定
#define		MCX514_CMD0C_SM		0x000C		// ソフトリミット－ 設定
#define		MCX514_CMD0D_AO		0x000D		// 加速カウンタオフセット 設定
#define		MCX514_CMD0E_LX		0x000E		// 論理位置カウンタ最大値 設定
#define		MCX514_CMD0F_RX		0x000F		// 実位置カウンタ最大値 設定
#define		MCX514_CMD10_MR0	0x0010		// 多目的レジスタ 0 設定
#define		MCX514_CMD11_MR1	0x0011		// 多目的レジスタ 1 設定
#define		MCX514_CMD12_MR2	0x0012		// 多目的レジスタ 2 設定
#define		MCX514_CMD13_MR3	0x0013		// 多目的レジスタ 3 設定
#define		MCX514_CMD14_HV		0x0014		// 原点検出速度 設定
#define		MCX514_CMD15_IV		0x0015		// 速度増減値 設定
#define		MCX514_CMD16_TM		0x0016		// タイマー値 設定
#define		MCX514_CMD17_SP1	0x0017		// スプリットパルス設定１
#define		MCX514_CMD18_SP2	0x0018		// スプリットパルス設定２
#define		MCX514_CMD19_TX		0x0019		// 補間・終点最大値設定
#define		MCX514_CMD1A_HLN	0x001A		// ヘリカル回転数設定
#define		MCX514_CMD1B_HLV	0x001B		// ヘリカル演算値設定

//////////////////////////////
// モード書き込み命令
//////////////////////////////
#define		MCX514_CMD20_MRM	0x0020		// 多目的レジスタモード設定
#define		MCX514_CMD21_P1M	0x0021		// PIO 信号設定１
#define		MCX514_CMD22_P2M	0x0022		// PIO 信号設定２･その他設定
#define		MCX514_CMD23_H1M	0x0023		// 自動原点出しモード設定１
#define		MCX514_CMD24_H2M	0x0024		// 自動原点出しモード設定２
#define		MCX514_CMD25_FLM	0x0025		// 入力信号フィルタモード設定
#define		MCX514_CMD26_SOM	0x0026		// 同期動作 SYNC0 設定
#define		MCX514_CMD27_S1M	0x0027		// 同期動作 SYNC1 設定
#define		MCX514_CMD28_S2M	0x0028		// 同期動作 SYNC2 設定
#define		MCX514_CMD29_S3M	0x0029		// 同期動作 SYNC3 設定
#define		MCX514_CMD2A_IPM	0x002A		// 補間モード設定

//////////////////////////////
// データ読み出し命令
//////////////////////////////
#define		MCX514_CMD30_LP		0x0030		// 論理位置カウンタ 読み出し
#define		MCX514_CMD31_RP		0x0031		// 実位置カウンタ 読み出し
#define		MCX514_CMD32_CV		0x0032		// 現在ドライブ速度 読み出し
#define		MCX514_CMD33_CA		0x0033		// 現在加減速度 読み出し
#define		MCX514_CMD34_MR0	0x0034		// 多目的レジスタ 0 読み出し
#define		MCX514_CMD35_MR1	0x0035		// 多目的レジスタ 1 読み出し
#define		MCX514_CMD36_MR2	0x0036		// 多目的レジスタ 2 読み出し
#define		MCX514_CMD37_MR3	0x0037		// 多目的レジスタ 3 読み出し
#define		MCX514_CMD38_CT		0x0038		// 現在タイマー値 読み出し
#define		MCX514_CMD39_TX		0x0039		// 補間・終点最大値 読み出し
#define		MCX514_CMD3A_CHLN	0x003A		// 現在ヘリカル回転数 読み出し
#define		MCX514_CMD3B_HLV	0x003B		// ヘリカル演算値 読み出し
```

```
#define             MCX514_CMD3D_WR1            0x003D          // WR1 設定値 読み出し
#define             MCX514_CMD3E_WR2            0x003E          // WR2 設定値 読み出し
#define             MCX514_CMD3F_WR3            0x003F          // WR3 設定値 読み出し
#define             MCX514_CMD40_MRM            0x0040          // 多目的レジスタモード設定 読み出し
#define             MCX514_CMD41_P1M            0x0041          // PIO 信号設定 1 読み出し
#define             MCX514_CMD42_P2M            0x0042          // PIO 信号設定 2･その他設定 読み出し
#define             MCX514_CMD43_AC             0x0043          // 加速度設定値 読み出し
#define             MCX514_CMD44_SV             0x0044          // 初速度設定値 読み出し
#define             MCX514_CMD45_DV             0x0045          // ドライブ速度設定値 読み出し
#define             MCX514_CMD46_TP             0x0046          // 移動パルス数/終点設定値 読み出し
#define             MCX514_CMD47_SP1            0x0047          // スプリットパルス設定 1 読み出し
#define             MCX514_CMD48_UI             0x0048          // 汎用入力値 読み出し

////////////////////////////
// ドライブ命令
////////////////////////////
#define             MCX514_CMD50_DRVRL          0x0050          // 相対位置ドライブ
#define             MCX514_CMD51_DRVNR          0x0051          // 反相対位置ドライブ
#define             MCX514_CMD52_DRVVP          0x0052          // ＋方向連続パルスドライブ
#define             MCX514_CMD53_DRVVM          0x0053          // －方向連続パルスドライブ
#define             MCX514_CMD54_DRVAB          0x0054          // 絶対位置ドライブ
#define             MCX514_CMD56_DRVSBRK        0x0056          // ドライブ減速停止
#define             MCX514_CMD57_DRVFBRK        0x0057          // ドライブ即停止
#define             MCX514_CMD58_DIRCP          0x0058          // 方向信号＋設定
#define             MCX514_CMD59_DIRCM          0x0059          // 方向信号－設定
#define             MCX514_CMD5A_HMSRC          0x005A          // 自動原点出し実行

////////////////////////////
// 補間命令
////////////////////////////
#define             MCX514_CMD60_LHK1           0x0060          // 1 軸直線補間ドライブ(マルチチップ用)
#define             MCX514_CMD61_LHK2           0x0061          // 2 軸直線補間ドライブ
#define             MCX514_CMD62_LHK3           0x0062          // 3 軸直線補間ドライブ
#define             MCX514_CMD63_LHK4           0x0063          // 4 軸直線補間ドライブ
#define             MCX514_CMD64_CHKCW          0x0064          // CW 円弧補間ドライブ
#define             MCX514_CMD65_CHKCCW         0x0065          // CCW 円弧補間ドライブ
#define             MCX514_CMD66_BHK2           0x0066          // 2 軸ビットパターン補間ドライブ
#define             MCX514_CMD67_BHK3           0x0067          // 3 軸ビットパターン補間ドライブ
#define             MCX514_CMD68_BHK4           0x0068          // 4 軸ビットパターン補間ドライブ
#define             MCX514_CMD69_HLCW           0x0069          // CW ヘリカル補間ドライブ
#define             MCX514_CMD6A_HLCCW          0x006A          // CCW ヘリカル補間ドライブ
#define             MCX514_CMD6B_HLPCW          0x006B          // CW ヘリカル演算
#define             MCX514_CMD6C_HLPCCW         0x006C          // CCW ヘリカル演算
#define             MCX514_CMD6D_DECEN          0x006D          // 減速有効
#define             MCX514_CMD6E_DECDIS         0x006E          // 減速無効
#define             MCX514_CMD6F_CLRSTEP        0x006F          // 補間割り込みクリア／補間ステップ

////////////////////////////
// 同期動作操作命令
////////////////////////////
#define             MCX514_CMD81_SYNC0EN        0x0081          // 同期動作 SYNC0 有効設定
#define             MCX514_CMD82_SYNC1EN        0x0082          // 同期動作 SYNC1 有効設定
#define             MCX514_CMD84_SYNC2EN        0x0084          // 同期動作 SYNC2 有効設定
#define             MCX514_CMD88_SYNC3EN        0x0088          // 同期動作 SYNC3 有効設定
#define             MCX514_CMD91_SYNC0DIS       0x0091          // 同期動作 SYNC0 無効設定
#define             MCX514_CMD92_SYNC1DIS       0x0092          // 同期動作 SYNC1 無効設定
#define             MCX514_CMD94_SYNC2DIS       0x0094          // 同期動作 SYNC2 無効設定
#define             MCX514_CMD98_SYNC3DIS       0x0098          // 同期動作 SYNC3 無効設定
#define             MCX514_CMDA1_SYNC0ACT       0x00A1          // 同期動作 SYNC0 起動
#define             MCX514_CMDA2_SYNC1ACT       0x00A2          // 同期動作 SYNC1 起動
#define             MCX514_CMDA4_SYNC2ACT       0x00A4          // 同期動作 SYNC2 起動
#define             MCX514_CMDA8_SYNC3ACT       0x00A8          // 同期動作 SYNC3 起動

////////////////////////////
// その他命令
////////////////////////////
#define             MCX514_CMD70_VINC           0x0070          // 速度増加
#define             MCX514_CMD71_VDEC           0x0071          // 速度減少
#define             MCX514_CMD72_DCC            0x0072          // 偏差カウンタクリア出力
#define             MCX514_CMD73_TMSTA          0x0073          // タイマー始動
```

```
#define             MCX514_CMD74_TMSTP          0x0074              // タイマー停止
#define             MCX514_CMD75_SPSTA          0x0075              // スプリットパルス開始
#define             MCX514_CMD76_SPSTP          0x0076              // スプリットパルス停止
#define             MCX514_CMD77_DHOLD          0x0077              // ドライブ開始ホールド
#define             MCX514_CMD78_DFREE          0x0078              // ドライブ開始フリー
#define             MCX514_CMD79_R2CLR          0x0079              // エラー･終了ステータスクリア
#define             MCX514_CMD7A_RR3P0          0x007A              // RR3 ページ０表示
#define             MCX514_CMD7B_RR3P1          0x007B              // RR3 ページ１表示
#define             MCX514_CMD1F_NOP            0x001F              // NOP
#define             MCX514_CMDFF_RST            0x00FF              // コマンドリセット

////////////////////////////////////////////////////////////////////////////
// 軸の定義
////////////////////////////////////////////////////////////////////////////
#define   MCX514_AXIS_X       0x01                                  // X 軸
#define   MCX514_AXIS_Y       0x02                                  // Y 軸
#define   MCX514_AXIS_Z       0x04                                  // Z 軸
#define   MCX514_AXIS_U       0x08                                  // U 軸
#define   MCX514_AXIS_ALL     0x0f                                  // 全軸指定
#define   MCX514_AXIS_NONE    0x00                                  // 軸指定なし

////////////////////////////////////////////////////////////////////////////
// アドレスの定義
////////////////////////////////////////////////////////////////////////////
#define   REG_ADDR            0x0000                                // ベースアドレス

// ライトレジスタ、リードレジスタの定義
#define   MCX514_WR0          0x00
#define   MCX514_WR1          0x02
#define   MCX514_WR2          0x04
#define   MCX514_WR3          0x06
#define   MCX514_WR4          0x08
#define   MCX514_WR5          0x0a
#define   MCX514_WR6          0x0c
#define   MCX514_WR7          0x0e
#define   MCX514_RR0          0x00
#define   MCX514_RR1          0x02
#define   MCX514_RR2          0x04
#define   MCX514_RR3          0x06
#define   MCX514_RR4          0x08
#define   MCX514_RR5          0x0a
#define   MCX514_RR6          0x0c
#define   MCX514_RR7          0x0e

unsigned short reg_read (unsigned short n);

#define   reg_write(n,c)      (*(volatile unsigned short *)n = ((volatile)c))
#define   reg_read(n)         (*(volatile unsigned short *)n)

////////////////////////////////////////////////////////////////////////////
// 共通関数の宣言
////////////////////////////////////////////////////////////////////////////
int WriteReg(volatile unsigned short *Adr, unsigned short Data);      // WR レジスタ書き込み共通関数
int ReadReg(volatile unsigned short *Adr, unsigned short *Data);      // RR レジスタ読み出し共通関数
int SetData(unsigned short Cmd, int Axis, long Data);                 // データ書き込み命令共通関数
int SetModeData(unsigned short Cmd, int Axis, unsigned short Data);   // モード書き込み命令共通関数
int GetData(unsigned short Cmd, int Axis, long *Data);                // データ読み出し命令共通関数
int ExeCmd(unsigned short Cmd, int Axis);                             // 命令実行共通関数

////////////////////////////////////////////////////////////////////////////
// WR レジスタ書き込み関数
////////////////////////////////////////////////////////////////////////////
int WriteReg0(unsigned short Data){                                   // WR0 レジスタ書き込み
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR0), Data));
}
int WriteReg1(int Axis, unsigned short Data){                         // WR1 レジスタ書き込み
          WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));                // 軸指定
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR1), Data));
}
```

```
int WriteReg2(int Axis, unsigned short Data){                          // WR2 レジスタ書き込み
          WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));                 // 軸指定
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR2), Data));
}
int WriteReg3(int Axis, unsigned short Data){                          // WR3 レジスタ書き込み
          WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));                 // 軸指定
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR3), Data));
}
int WriteReg4(unsigned short Data){                                    // WR4 レジスタ書き込み
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR4), Data));
}
int WriteReg6(unsigned short Data){                                    // WR6 レジスタ書き込み
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR6), Data));
}
int WriteReg7(unsigned short Data){                                    // WR7 レジスタ書き込み
          return(WriteReg((volatile unsigned short*)(REG_ADDR + MCX514_WR7), Data));
}

////////////////////////////////////////////////////////////////////////////////
// RR レジスタ読み出し関数
////////////////////////////////////////////////////////////////////////////////
int ReadReg0(unsigned short *Data){                                    // RR0 レジスタ読み出し
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR0), Data));
}
int ReadReg1(int Axis, unsigned short *Data){                          // RR1 レジスタ読み出し
          WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));                 // 軸指定
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR1), Data));
}
int ReadReg2(int Axis, unsigned short *Data){                          // RR2 レジスタ読み出し
          WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));                 // 軸指定
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR2), Data));
}
int ReadReg3(int Page, int Axis, unsigned short *Data){                // RR3 レジスタ読み出し
          if (Page == 0) {                                             // ページ0指定
                    WriteReg0(((Axis << 8) + MCX514_CMD7A_RR3P0));
          }
          else {                                                       // ページ1指定
                    WriteReg0(((Axis << 8) + MCX514_CMD7B_RR3P1));
          }
          WriteReg0(((Axis << 8) + MCX514_CMD1F_NOP));                 // 軸指定
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR3), Data));
}
int ReadReg4(unsigned short *Data){                                    // RR4 レジスタ読み出し
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR4), Data));
}
int ReadReg5(unsigned short *Data){                                    // RR5 レジスタ読み出し
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR5), Data));
}
int ReadReg6(unsigned short *Data){                                    // RR6 レジスタ読み出し
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR6), Data));
}
int ReadReg7(unsigned short *Data){                                    // RR7 レジスタ読み出し
          return(ReadReg((volatile unsigned short*)(REG_ADDR + MCX514_RR7), Data));
}


////////////////////////////////////////////////////////////////////////////////
// データ書き込み命令 関数
////////////////////////////////////////////////////////////////////////////////
int SetStartSpd(int Axis, long Data){                                  // 初速度 設定
          return(SetData(MCX514_CMD04_SV, Axis, Data));
}
int SetSpeed(int Axis, long Data){                                     // ドライブ速度 設定
          return(SetData(MCX514_CMD05_DV, Axis, Data));
}
int SetJerk(int Axis, long Data){                                      // 加速度増加率 設定
          return(SetData(MCX514_CMD00_JK, Axis, Data));
}
```

```
int SetDJerk(int Axis, long Data){                                      // 減速度増加率 設定
        return(SetData(MCX514_CMD01_DJ, Axis, Data));
}
int SetAcc(int Axis, long Data){                                        // 加速度 設定
        return(SetData(MCX514_CMD02_AC, Axis, Data));
}
int SetDec(int Axis, long Data){                                        // 減速度 設定
        return(SetData(MCX514_CMD03_DC, Axis, Data));
}
int SetPulse(int Axis, long Data){                                      // 移動パルス数/終点 設定
        return(SetData(MCX514_CMD06_TP, Axis, Data));
}

int SetDecP(int Axis, long Data){                                       // マニュアル減速点 設定
        return(SetData(MCX514_CMD07_DP, Axis, Data));
}
int SetLp(int Axis, long Data){                                         // 論理位置カウンタ 設定
        return(SetData(MCX514_CMD09_LP, Axis, Data));
}
int SetRp(int Axis, long Data){                                         // 実位置カウンタ 設定
        return(SetData(MCX514_CMD0A_RP, Axis, Data));
}
int SetCompP(int Axis, long Data){                                      // ソフトリミット＋ 設定
        return(SetData(MCX514_CMD0B_SP, Axis, Data));
}
int SetCompM(int Axis, long Data){                                      // ソフトリミットー 設定
        return(SetData(MCX514_CMD0C_SM, Axis, Data));
}
int SetAccOfst(int Axis, long Data){                                    // 加速カウンタオフセット 設定
        return(SetData(MCX514_CMD0D_AO, Axis, Data));
}
int SetHomeSpd(int Axis, long Data){                                    // 原点検出速度 設定
        return(SetData(MCX514_CMD14_HV, Axis, Data));
}
int SetLpMax(int Axis, long Data){                                      // 論理位置カウンタ最大値 設定
        return(SetData(MCX514_CMD0E_LX, Axis, Data));
}
int SetRpMax(int Axis, long Data){                                      // 実位置カウンタ最大値 設定
        return(SetData(MCX514_CMD0F_RX, Axis, Data));
}
int SetMR0(int Axis, long Data){                                        // 多目的レジスタ 0 設定
        return(SetData(MCX514_CMD10_MR0, Axis, Data));
}
int SetMR1(int Axis, long Data){                                        // 多目的レジスタ 1 設定
        return(SetData(MCX514_CMD11_MR1, Axis, Data));
}
int SetMR2(int Axis, long Data){                                        // 多目的レジスタ 2 設定
        return(SetData(MCX514_CMD12_MR2, Axis, Data));
}
int SetMR3(int Axis, long Data){                                        // 多目的レジスタ 3 設定
        return(SetData(MCX514_CMD13_MR3, Axis, Data));
}
int SetSpeedInc(int Axis, long Data){                                   // 速度増減値 設定
        return(SetData(MCX514_CMD15_IV, Axis, Data));
}
int SetTimer(int Axis, long Data){                                      // タイマー値 設定
        return(SetData(MCX514_CMD16_TM, Axis, Data));
}
int SetSplit1(int Axis, unsigned short Data1, unsigned short Data2) {   // スプリットパルス設定 1
        long Data;

        Data = ((Data1 << 16) | Data2);
        return(SetData(MCX514_CMD17_SP1, Axis, Data));
}
int SetSplit2(int Axis, long Data){                                     // スプリットパルス設定 2
        return(SetData(MCX514_CMD18_SP2, Axis, Data));
}
int SetTPMax(long Data){                                                // 補間・終点最大値設定
        return(SetData(MCX514_CMD39_TX, MCX514_AXIS_NONE, Data));
}
```

```
int SetHLNumber(unsigned short Data){                                   // ヘリカル回転数設定
        return(SetData(MCX514_CMD3A_CHLN, MCX514_AXIS_NONE, (long )Data));
}
int SetHLValue(long Data){                                              // ヘリカル演算値設定
        return(SetData(MCX514_CMD3B_HLV, MCX514_AXIS_NONE, Data));
}

////////////////////////////////////////////////////////////////////////////////
// モード書き込み命令 関数
////////////////////////////////////////////////////////////////////////////////
int SetModeMRm(int Axis, unsigned short Data){                          // 多目的レジスタモード設定
        return(SetModeData(MCX514_CMD20_MRM, Axis, Data));
}
int SetModePIO1(int Axis, unsigned short Data){                         // PIO 信号設定１
        return(SetModeData(MCX514_CMD21_P1M, Axis, Data));
}
int SetModePIO2(int Axis, unsigned short Data){                         // PIO 信号設定２･その他設定
        return(SetModeData(MCX514_CMD22_P2M, Axis, Data));
}
int SetModeHMSrch1(int Axis, unsigned short Data){                      // 自動原点出しモード設定 1
        return(SetModeData(MCX514_CMD23_H1M, Axis, Data));
}
int SetModeHMSrch2(int Axis, unsigned short Data){                      // 自動原点出しモード設定 2
        return(SetModeData(MCX514_CMD24_H2M, Axis, Data));
}
int SetModeFilter(int Axis, unsigned short Data){                       // 入力信号フィルタモード設定
        return(SetModeData(MCX514_CMD25_FLM, Axis, Data));
}
int SetModeSync0(int Axis, unsigned short Data){                        // 同期動作 SYNC0 設定
        return(SetModeData(MCX514_CMD26_SOM, Axis, Data));
}
int SetModeSync1(int Axis, unsigned short Data){                        // 同期動作 SYNC1 設定
        return(SetModeData(MCX514_CMD27_S1M, Axis, Data));
}
int SetModeSync2(int Axis, unsigned short Data){                        // 同期動作 SYNC2 設定
        return(SetModeData(MCX514_CMD28_S2M, Axis, Data));
}
int SetModeSync3(int Axis, unsigned short Data){                        // 同期動作 SYNC3 設定
        return(SetModeData(MCX514_CMD29_S3M, Axis, Data));
}
int SetModeIPM(unsigned short Data){                                    // 補間モード設定
        return(SetModeData(MCX514_CMD2A_IPM, MCX514_AXIS_NONE, Data));
}

////////////////////////////////////////////////////////////////////////////////
// データ読み出し関数
////////////////////////////////////////////////////////////////////////////////
int GetLp(int Axis, long *Data){                                        // 論理位置カウンタ 読み出し
        return(GetData(MCX514_CMD30_LP, Axis, Data));
}
int GetRp(int Axis, long *Data){                                        // 実位置カウンタ 読み出し
        return(GetData(MCX514_CMD31_RP, Axis, Data));
}
int GetCV(int Axis, long *Data){                                        // 現在ドライブ速度 読み出し
        return(GetData(MCX514_CMD32_CV, Axis, Data));
}
int GetCA(int Axis, long *Data){                                        // 現在加減速度 読み出し
        return(GetData(MCX514_CMD33_CA, Axis, Data));
}
int GetCT(int Axis, long *Data){                                        // 現在タイマー値 読み出し
        return(GetData(MCX514_CMD38_CT, Axis, Data));
}
int GetMR0(int Axis, long *Data){                                       // 多目的レジスタ 0 読み出し
        return(GetData(MCX514_CMD34_MR0, Axis, Data));
}
int GetMR1(int Axis, long *Data){                                       // 多目的レジスタ 1 読み出し
        return(GetData(MCX514_CMD35_MR1, Axis, Data));
}
```

```
int GetMR2(int Axis, long *Data){                                       // 多目的レジスタ 2 読み出し
        return(GetData(MCX514_CMD36_MR2, Axis, Data));
}
int GetMR3(int Axis, long *Data){                                       // 多目的レジスタ 3 読み出し
        return(GetData(MCX514_CMD37_MR3, Axis, Data));
}
int GetTX(long *Data){                                                  // 補間・終点最大値 読み出し
        return(GetData(MCX514_CMD39_TX, MCX514_AXIS_NONE, Data));
}
int GetCHLN(long *Data){                                                // 現在ヘリカル回転数 読み出し
        return(GetData(MCX514_CMD3A_CHLN, MCX514_AXIS_NONE, Data));
}
int GetHLV(long *Data){                                                 // ヘリカル演算値 読み出し
        return(GetData(MCX514_CMD3B_HLV, MCX514_AXIS_NONE, Data));
}

int GetWR1(int Axis, long *Data){                                       // WR1 設定値 読み出し
        return(GetData(MCX514_CMD3D_WR1, Axis, Data));
}
int GetWR2(int Axis, long *Data){                                       // WR2 設定値 読み出し
        return(GetData(MCX514_CMD3E_WR2, Axis, Data));
}
int GetWR3(int Axis, long *Data){                                       // WR3 設定値 読み出し
        return(GetData(MCX514_CMD3F_WR3, Axis, Data));
}
int GetMRM(int Axis, long *Data){                                       // 多目的レジスタモード設定 読み出し
        return(GetData(MCX514_CMD40_MRM, Axis, Data));
}
int GetP1M(int Axis, long *Data){                                       // PIO 信号設定 1 読み出し
        return(GetData(MCX514_CMD41_P1M, Axis, Data));
}
int GetP2M(int Axis, long *Data){                                       // PIO 信号設定 2･その他設定 読み出し
        return(GetData(MCX514_CMD42_P2M, Axis, Data));
}
int GetAc(int Axis, long *Data ){                                       // 加速度設定値 読み出し
        return(GetData(MCX514_CMD43_AC, Axis, Data));
}
int GetStartSpd(int Axis, long *Data ){                                 // 初速度設定値 読み出し
        return(GetData(MCX514_CMD44_SV, Axis, Data));
}
int GetSpeed(int Axis, long *Data ){                                    // ドライブ速度設定値 読み出し
        return(GetData(MCX514_CMD45_DV, Axis, Data));
}
int GetPulse(int Axis, long *Data ){                                    // 移動パルス数/終点設定値 読み出し
        return(GetData(MCX514_CMD46_TP, Axis, Data));
}
int GetSplit(int Axis, long *Data ){                                    // スプリットパルス設定 1 読み出し
        return(GetData(MCX514_CMD47_SP1, Axis, Data));
}
int GetUI(long *Data ){                                                 // 汎用入力値 読み出し
        return(GetData(MCX514_CMD48_UI, MCX514_AXIS_NONE, Data));
}

////////////////////////////////////////////////////////////////////////////////
// ドライブ命令 関数
////////////////////////////////////////////////////////////////////////////////
int ExeDRVRL(int Axis){                                                 // 相対位置ドライブ
        return (ExeCmd(MCX514_CMD50_DRVRL, Axis));
}
int ExeDRVNR(int Axis){                                                 // 反相対位置ドライブ
        return (ExeCmd(MCX514_CMD51_DRVNR, Axis));
}
int ExeDRVVP(int Axis){                                                 // ＋方向連続パルスドライブ
        return (ExeCmd(MCX514_CMD52_DRVVP, Axis));
}
int ExeDRVVM(int Axis){                                                 // －方向連続パルスドライブ
        return (ExeCmd(MCX514_CMD53_DRVVM, Axis));
}
```

```
int ExeDRVAB(int Axis){                                                 // 絶対位置ドライブ
        return (ExeCmd(MCX514_CMD54_DRVAB, Axis));
}
int ExeDRVSBRK(int Axis){                                               // ドライブ減速停止
        return (ExeCmd(MCX514_CMD56_DRVSBRK, Axis));
}
int ExeDRVFBRK(int Axis){                                               // ドライブ即停止
        return (ExeCmd(MCX514_CMD57_DRVFBRK, Axis));
}
int ExeDIRCP(int Axis){                                                 // 方向信号＋設定
        return (ExeCmd(MCX514_CMD58_DIRCP, Axis));
}
int ExeDIRCM(int Axis){                                                 // 方向信号－設定
        return (ExeCmd(MCX514_CMD59_DIRCM, Axis));
}
int ExeHMSRC(int Axis){                                                 // 自動原点出し実行
        return (ExeCmd(MCX514_CMD5A_HMSRC, Axis));
}

////////////////////////////////////////////////////////////////////////////////
// 補間命令 関数
////////////////////////////////////////////////////////////////////////////////
int ExeLHK1(void ){                                                     // 1軸直線補間ドライブ(マルチチップ用)
        return (ExeCmd(MCX514_CMD60_LHK1, MCX514_AXIS_NONE));
}
int ExeLHK2(void ){                                                     // 2軸直線補間ドライブ
        return (ExeCmd(MCX514_CMD61_LHK2, MCX514_AXIS_NONE));
}
int ExeLHK3(void ){                                                     // 3軸直線補間ドライブ
        return (ExeCmd(MCX514_CMD62_LHK3, MCX514_AXIS_NONE));
}
int ExeLHK4(void ){                                                     // 4軸直線補間ドライブ
        return (ExeCmd(MCX514_CMD63_LHK4, MCX514_AXIS_NONE));
}
int ExeCHKCW(void ){                                                    // CW円弧補間ドライブ
        return (ExeCmd(MCX514_CMD64_CHKCW, MCX514_AXIS_NONE));
}
int ExeCHKCCW(void ){                                                   // CCW円弧補間ドライブ
        return (ExeCmd(MCX514_CMD64_CHKCCW, MCX514_AXIS_NONE));
}
int ExeBHK2(void ){                                                     // 2軸ビットパターン補間ドライブ
        return (ExeCmd(MCX514_CMD66_BHK2, MCX514_AXIS_NONE));
}
int ExeBHK3(void ){                                                     // 3軸ビットパターン補間ドライブ
        return (ExeCmd(MCX514_CMD67_BHK3, MCX514_AXIS_NONE));
}
int ExeBHK4(void ){                                                     // 4軸ビットパターン補間ドライブ
        return (ExeCmd(MCX514_CMD68_BHK4, MCX514_AXIS_NONE));
}
int ExeHLCW(void ){                                                     // CWヘリカル補間ドライブ
        return (ExeCmd(MCX514_CMD69_HLCW, MCX514_AXIS_NONE));
}
int ExeHLCCW(void ){                                                    // CCWヘリカル補間ドライブ
        return (ExeCmd(MCX514_CMD6A_HLCCW, MCX514_AXIS_NONE));
}
int ExeHLPCW(void ){                                                    // CWヘリカル演算
        return (ExeCmd(MCX514_CMD6B_HLPCW, MCX514_AXIS_NONE));
}
int ExeHLPCCW(void ){                                                   // CCWヘリカル演算
        return (ExeCmd(MCX514_CMD6C_HLPCCW, MCX514_AXIS_NONE));
}
int ExeDECEN(void ){                                                    // 減速有効
        return (ExeCmd(MCX514_CMD6D_ DECEN, MCX514_AXIS_NONE));
}
int ExeDECDIS(void ){                                                   // 減速無効
        return (ExeCmd(MCX514_CMD6E_ DECDIS, MCX514_AXIS_NONE));
}
int ExeCLRSTEP(void ){                                                  // 補間割り込みクリア／補間ステップ
        return (ExeCmd(MCX514_CMD6F_CLRSTEP, MCX514_AXIS_NONE));
}
```

```
////////////////////////////////////////////////////////////////////////////////
// 同期動作操作命令 関数
////////////////////////////////////////////////////////////////////////////////
int ExeSYNC(int Axis, unsigned short Cmd){                              // 同期動作関連命令
          return (ExeCmd(Cmd, Axis));
}

////////////////////////////////////////////////////////////////////////////////
// その他命令 関数
////////////////////////////////////////////////////////////////////////////////
int ExeVINC(int Axis){                                                  // 速度増加
          return (ExeCmd(MCX514_CMD70_VINC, Axis));
}
int ExeVDEC(int Axis){                                                  // 速度減少
          return (ExeCmd(MCX514_CMD71_VDEC, Axis));
}
int ExeDCC(int Axis){                                                   // 偏差カウンタクリア出力
          return (ExeCmd(MCX514_CMD72_DCC, Axis));
}
int ExeTMSTA(int Axis){                                                 // タイマー始動
          return (ExeCmd(MCX514_CMD73_TMSTA, Axis));
}

int ExeTMSTP(int Axis){                                                 // タイマー停止
          return (ExeCmd(MCX514_CMD74_TMSTP, Axis));
}
int ExeSPSTA(int Axis){                                                 // スプリットパルス開始
          return (ExeCmd(MCX514_CMD75_SPSTA, Axis));
}
int ExeSPSTP(int Axis){                                                 // スプリットパルス停止
          return (ExeCmd(MCX514_CMD76_SPSTP, Axis));
}
int ExeDHOLD(int Axis){                                                 // ドライブ開始ホールド
          return (ExeCmd(MCX514_CMD77_DHOLD, Axis));
}
int ExeDFREE (int Axis){                                                // ドライブ開始フリー
          return (ExeCmd(MCX514_CMD78_DFREE, Axis));
}
int ExeR2CLR(int Axis){                                                 // エラー･終了ステータスクリア
          return (ExeCmd(MCX514_CMD79_R2CLR, Axis));
}
int ExeRR3P0(int Axis){                                                 // RR3 ページ 0 表示
          return (ExeCmd(MCX514_CMD7A_RR3P0, Axis));
}
int ExeRR3P1(int Axis){                                                 // RR3 ページ 1 表示
          return (ExeCmd(MCX514_CMD7B_RR3P1, Axis));
}
int ExeNOP(int Axis){                                                   // NOP
          return (ExeCmd(MCX514_CMD1F_NOP, Axis));
}
int ExeSRST(void ){                                                     // コマンドリセット
          return (ExeCmd(MCX514_CMDFF_RST, MCX514_AXIS_NONE));
}

////////////////////////////////////////////////////////////////////////////////
// 共通関数
////////////////////////////////////////////////////////////////////////////////
// WR レジスタ書き込み共通関数(I/O ポートアクセス部分。以下のプログラム例は、SH マイコンの例です。)
int WriteReg(volatile unsigned short *Adr, unsigned short Data) {
          reg_write(Adr, Data);
          return 0;
}
// RR レジスタ読み出し共通関数(I/O ポートアクセス部分。以下のプログラム例は、SH マイコンの例です。)
int ReadReg(volatile unsigned short *Adr, unsigned short *Data) {
          *Data = reg_read(Adr);
          return 0;
}
```

```
// データ書き込み共通関数
// WR6、WR7 へデータを書き込み、その後、WR0 に命令を書き込むことでデータ書き込みが行えます。
int SetData(unsigned short Cmd, int Axis, long Data) {

          long mask_data = 0x0000ffff;
          unsigned short write_data;

          // WR6 へのデータの下位 16bit を書き込む
          write_data = (unsigned short )(Data & mask_data);
          WriteReg6(write_data);

          // WR7 へのデータの上位 16bit を書き込む
          write_data = (unsigned short )(Data >> 16);
          WriteReg7(write_data);

          // 命令書き込み(WR0 への書き込み)
          WriteReg0(((Axis << 8) + Cmd));

          return 0;
}
// モード書き込み命令 共通関数
// WR6 へデータを書き込み、その後、WR0 に命令を書き込むことでデータ書き込みが行えます。
int SetModeData(unsigned short Cmd, int Axis, unsigned short Data) {

          // WR6 へのデータの下位 16bit を書き込む
          WriteReg6(Data);

          // 命令書き込み(WR0 への書き込み)
          WriteReg0(((Axis << 8) + Cmd));

          return 0;
}
// データ読み出し共通関数
// WR0 に命令を書き込み、その後、RR6、RR7 を読み出すことで、データ読み出しが行えます。
int GetData(unsigned short Cmd, int Axis, long *Data) {

          unsigned short rdata1,rdata2;
          long retdata = 0x00000000;

          if (Data == NULL) return 0;

          // 命令書き込み(WR0 への書き込み)
          WriteReg0(((Axis << 8) + Cmd));

          // RR7 読み出し
          ReadReg7(&rdata1);

          // RR6 読み出し
          ReadReg6(&rdata2);

          // 読み込みデータ作成
          retdata = (long )rdata1;                // 上位 16bit に、RR7 の値をセット
          *Data = (retdata << 16);
          retdata = (long )rdata2;                // 下位 16bit に、RR6 の値をセット
          *Data = *Data + retdata;

          return 0;
}
// 命令実行共通関数
int ExeCmd(unsigned short Cmd, int Axis) {

          // 命令書き込み(WR0 への書き込み)
          WriteReg0(((Axis << 8) + Cmd));

          return 0;
}
```

```
// ドライブ終了待ち
void waitdrive(int Axis) {

          unsigned short rrData;

          ReadReg0(&rrData);                        // RR0 の読み出し
          while ((rrData & Axis)) {                 // ドライブ中なら
                    ReadReg0(&rrData);              // RR0 の読み出し
          }
}

// スプリットパルス終了待ち
void waitsplit(int Axis) {

          unsigned short rrData;

          ReadReg3(1, Axis, &rrData);               // RR3 ページ１の読み出し
          while ((rrData & 0x0800)) {               // スプリットパルス動作中なら
                    ReadReg3(1, Axis, &rrData);     // RR3 ページ１の読み出し
          }
}

////////////////////////////////////////////////////////////////////////////
// 動作例関数
////////////////////////////////////////////////////////////////////////////
// 自動原点出し
// MCX514 マニュアルの「2.5.8 自動原点出し実例」の「例１　原点信号を用いた原点出し例」を行います。
void homesrch(void ) {

          WriteReg2(MCX514_AXIS_X, 0x0800);         // 原点信号論理設定 STOP1 Low アクティブ
                                                    // ハードリミット有効
          SetModeFilter(MCX514_AXIS_X, 0x0A0F);     // STOP1 フィルタ有効
                                                    // フィルタ遅延 512μsec
          SetModeHMSrch1(MCX514_AXIS_X, 0x8037);    // ステップ４ 実行
                                                    // ステップ３ 不実行
                                                    // ステップ２ 実行
                                                    //            検出信号 STOP1
                                                    //            検出方向 －方向
                                                    //            LP,RP クリア 無効
                                                    //            DCC クリア 無効
                                                    // ステップ１ 実行
                                                    //            検出信号 STOP1
                                                    //            検出方向 －方向
          SetModeHMSrch2(MCX514_AXIS_X, 0x0000);    // ステップ間タイマー 無効
                                                    // 原点出し終了時 LP,RP クリア 無効
          SetAcc(MCX514_AXIS_X, 95000);             // 加速度 95,000 pps/sec
          SetStartSpd(MCX514_AXIS_X, 1000);         // 初速度 1000pps
          SetSpeed(MCX514_AXIS_X, 20000);           // ステップ１、４の速度 20000pps
          SetHomeSpd(MCX514_AXIS_X, 500);           // ステップ２の速度 500pps
          SetPulse(MCX514_AXIS_X, 3500);            // オフセット移動パルス量 3500
          ExeHMSRC(MCX514_AXIS_X);                  // 自動原点出し実行
          waitdrive(MCX514_AXIS_X);                 // ドライブ終了待ち
}
// 全軸 S 字加減速ドライブ
void drive(void ) {

          SetStartSpd(MCX514_AXIS_ALL, 10);         // 初速度 10pps
          SetSpeed(MCX514_AXIS_ALL, 40000);         // ドライブ速度 40Kpps
          SetAcc(MCX514_AXIS_ALL, 536870911);       // 加速度 仕様最大
          SetJerk(MCX514_AXIS_ALL, 89300);          // 加速度増加率 89.3Kpps/sec2
          SetPulse(MCX514_AXIS_ALL, 70000);         // 移動パルス数 70000
          SetLp(MCX514_AXIS_ALL, 0);                // 論理位置カウンタ クリア
          WriteReg3(MCX514_AXIS_ALL, 0x0004);       // S 字加減速ドライブ指定
          ExeDRVRL(MCX514_AXIS_ALL);                // 相対位置ドライブ
          waitdrive(MCX514_AXIS_ALL);               // ドライブ終了待ち

}
```

```
// 同期動作
//MCX514 マニュアルの「2.6.6 同期動作の実例」の「例 3　ドライブ中に、指定位置 A から指定位置 B までを通過する時間を求める」を行います。
void sync(void ) {

        // 10kpps の定速ドライブ
        SetStartSpd(MCX514_AXIS_X,8000000);     // 初速度 8Mpps(仕様最大)
        SetSpeed(MCX514_AXIS_X,10000);          // ドライブ速度 10Kpps
        SetLp(MCX514_AXIS_X,0);                 // 論理位置カウンタ 0
        SetPulse(MCX514_AXIS_X,60000);          // 移動パルス数 60000
        SetMR0(MCX514_AXIS_X,10000);            // MR0 10000
        SetMR1(MCX514_AXIS_X,55000);            // MR1 55000
        SetTimer(MCX514_AXIS_X,2147483647);     // タイマー値(仕様最大)
        WriteReg1(MCX514_AXIS_X,0x2000);        // WR1　同期動作セット 1 起動
        SetModeMRm(MCX514_AXIS_X,0x0000);       // MR0 を LP と比較。比較条件≧
                                                // MR1 を LP と比較。比較条件≧
        SetModeSync0(MCX514_AXIS_X,0x0151);     // SYNC0 設定
                                                // 起動要因　MRm 比較が真に変化した
                                                // 動作　タイマー始動
        SetModeSync1(MCX514_AXIS_X,0x0071);     // SYNC1 設定
                                                // 起動要因　MRm 比較が真に変化した
                                                // 動作　セーブ　CT→MRm
        ExeSYNC(MCX514_AXIS_X, (MCX514_CMD81_SYNC0EN | MCX514_CMD82_SYNC1EN));
                                                // SYNC0,1 有効
        ExeDRVRL(MCX514_AXIS_X);                // 相対位置ドライブ
        waitdrive(MCX514_AXIS_X);               // ドライブ終了待ち
}

// スプリットパルス
// MCX514 マニュアルの「2.7.6 スプリットパルスの実例」の「例 1　ドライブ開始からスプリットパルスを開始」を行います。
void split(void ) {

        // 1000pps の定速ドライブ
        SetStartSpd(MCX514_AXIS_X,8000000);     // 初速度 8Mpps(仕様最大)
        SetSpeed(MCX514_AXIS_X,1000);           // ドライブ速度 1000pps
        SetLp(MCX514_AXIS_X,0);                 // 論理位置カウンタ
        SetSplit1(MCX514_AXIS_X,5,9);           // スプリット長 9　パルス幅 5
        SetSplit2(MCX514_AXIS_X,10);            // パルス数　10
        SetModePIO2(MCX514_AXIS_X,0x0800);      // パルス論理正　開始パルス有
        ExeSPSTA(MCX514_AXIS_X);                // スプリットパルス開始
        ExeDRVVP(MCX514_AXIS_X);                // ＋方向連続パルスドライブ
        waitsplit(MCX514_AXIS_X);               // スプリットパルス終了待ち
        ExeDRVFBRK(MCX514_AXIS_X);              // ドライブ即停止
        waitdrive(MCX514_AXIS_X);               // ドライブ終了待ち
}

// メイン関数
void main(void ) {

        ExeSRST();                              // コマンドリセット

        homesrch();                             // 自動原点出し

        drive();                                // S 字加減速ドライブ

        sync();                                 // 同期動作

        split();                                // スプリットパルス

}
```

10. 電気的特性

10.1 DC 特性

■ 絶対最大定格

項　目	記号	条件	定　格	単位
電源電圧	V_{DD}	−	-0.3 ～ +4.0	V
入力電圧	V_I	$V_I < V_{DD}+3.0V$	-0.3 ～ +7.0	V
出力電圧	V_O	$V_O < V_{DD}+3.0V$	-0.3 ～ +7.0	V
出力電流	I_O	−	±30	mA
保存温度	T_{STG}		-65 ～ +150	℃

■ 推奨動作条件

項　目	記号	定　格	単位
電源電圧	V_{DD}	3.3±0.3	V
動作温度	Ta	-40 ～ +85	℃

■ DC特性

（ Ta = -40 ～ +85℃, V_{DD} = 3.3v±0.3V ）

項　目	記号	条　件	最小	標準	最大	単位	備　考
高レベル入力電圧	V_{IH}		2.0		5.5	V	
低レベル入力電圧	V_{IL}		-0.3		0.8	V	
高レベル入力電流	I_{IH}	$V_{IN} = V_{DD}$			1.0	μA	
低レベル入力電流	I_{IL}	V_{IN} = 0V			-1.0	μA	
高レベル出力電圧	V_{OH}	I_{OH} = 0mA	V_{DD}-0.2			V	注1
		I_{OH} = -12mA	V_{DD}-0.4			V	D15～D0 信号
		I_{OH} = -6mA	V_{DD}-0.4			V	上記以外の信号
低レベル出力電圧	V_{OL}	I_{OL} = 0mA			0.1	V	
		I_{OL} = 12mA			0.4	V	D15～D0 信号 INT0N, INT1N 信号
		I_{OL} = 6mA			0.4	V	上記以外の信号
出力リーク電流	I_{OZ}	$V_{OUT}=V_{DD}$ or GND	-1		1	μA	D15～D0 信号 PIN6, PIN5 信号 INT0N, ITN1N 信号 SDA 信号
シュミットトリガ ヒステリシス電圧	V_H		0.1			V	
消費電流	I_{DD}	I_{IO}=0mA, CLK=16MHz		150	204	mA	
		I_{IO}=0mA, CLK=20MHz		190	252		

注1:INT0N, INT1N 出力信号および PIN6, 5、SDA 信号は、オープンドレイン出力ですので、高レベル出力電圧の項目はありません。

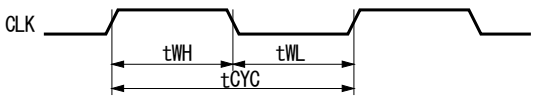
■ 端子容量

項　目	記号	条　件	最小	標準	最大	単位	備　考
入出力容量	C_{IO}	Ta=25℃, f=1MHz			10	pF	D15～D0 信号 nPI07～nPI00 信号 PIN7～PIN0 信号 SDA 信号
入力容量	C_I				10	pF	その他の入力端子

10.2 AC 遅延特性 （ Ta = −40～+85℃, V_{DD} = +3.3V±10%, 出力負荷条件:D15～D0,INTN:85pF,SDA:400pF,その他:50pF ）

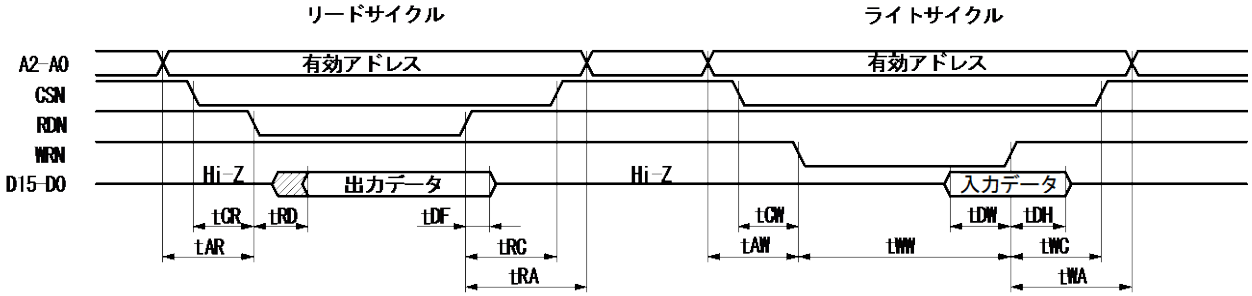
10.2.1 クロック

■ ＣＬＫ入力信号



記号	項　　目	最小	標準	最大	単位
tCYC	CLK 周期	50	62.5		nS
tWH	CLK Hi レベル幅	15			nS
tWL	CLK Low レベル幅	15			nS

10.2.2 CPU リード／ライトサイクル

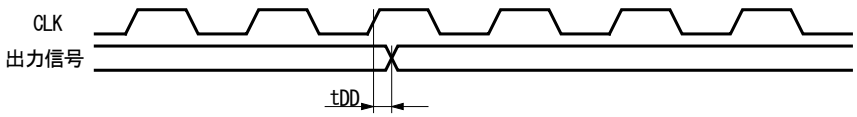


上図は、16ビットデータバス(H16L8=Hi)のときの信号です。8ビットデータバス(H16L8=Low)のときは、図においてアドレス信号がA3～A0、データ信号がD7～D0になります。

記号	項　　目	最小	最大	単位
tAR	アドレスセットアップ時間 （ to RDN↓）	0		nS
tCR	CSN セットアップ時間 （ to RDN↓）	0		nS
tRD	出力データ遅延時間 (from RDN↓)		21	nS
tDF	出力データ保持時間 (from RDN↑)	0	12	nS
tRC	CSN 保持時間 (from RDN↑)	0		nS
tRA	アドレス保持時間 (from RDN↑)	3		nS
tAW	アドレスセットアップ時間 （ to WRN↓）	0		nS
tCW	CSN セットアップ時間 （ to WRN↓）	0		nS
tWW	WRN Low レベルパルス幅	30		nS
tDW	入力データセットアップ時間 （ to WRN↑）	10		nS
tDH	入力データ保持時間 (from WRN↑)	0		nS
tWC	CSN 保持時間 (from WRN↑)	0		nS
tWA	アドレス保持時間 (from WRN↑)	4		nS

10.2.3 CLK／出力信号遅延

次の出力信号は、常に、CLK信号に同期しています。CLKの↑でレベルが変化します。



出力信号:nPP, nPM, nDCC, nSPLTP, nPI07～0(機能選択による)

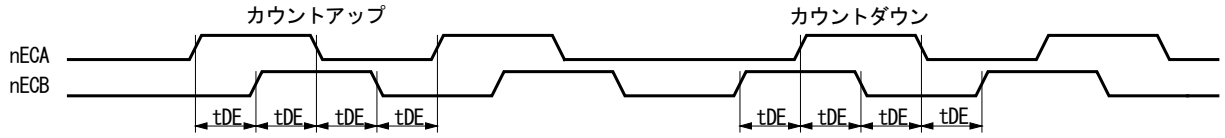
記号	項　　目	最小	最大	単位
tDD	CLK↑ → 出力信号↑↓ 遅延時間	7	30	nS

出力信号:INT0N, INT1N

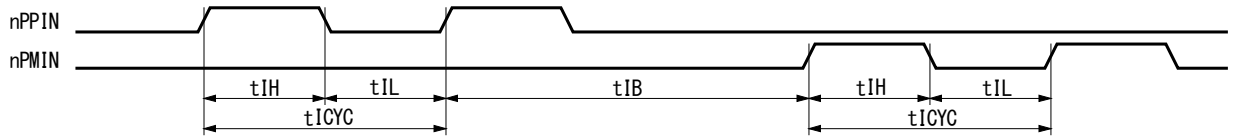
記号	項　　目	最小	最大	単位
tDD	CLK↑ → INTON,INT1N信号↓ 遅延時間	12	22	nS

10.2.4 入力パルス

■ 2相パルス入力モード



■ アップダウンパルス入力モード



a. 2相パルス入力モードでは、nECA, nECB入力が変化すると、実位置カウンタは、最大CLK4サイクル後に変化後の値になります。

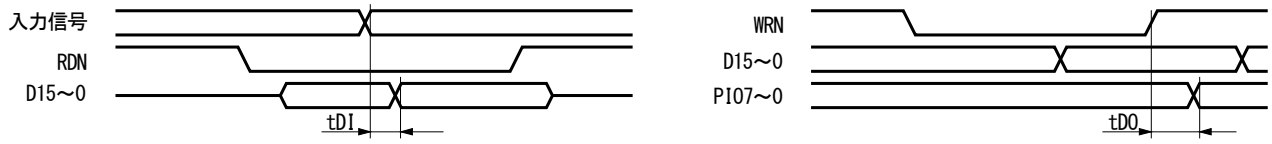
b. アップダウンパルス入力モードでは、nPPIN, nPMIN 入力の↑から最大 CLK4サイクル後に、実位置カウンタは変化後の値になります。

記号	項　　目	最小	最大	単位
tDE	nECA, nECB 位相差時間	tCYC +20		nS
tIH	nPPIN, nPMIN Hi レベル幅	tCYC +20		nS
tIL	nPPIN, nPMIN Low レベル幅	tCYC +20		nS
tICYC	nPPIN, nPMIN 周期	tCYC×2 +20		nS
tIB	nPPIN↑ ⟷ nPMIN↑時間	tCYC×2 +20		nS

tCYC は CLK の周期です。

10.2.5 汎用入／出力信号(nPIO7～0)

左下図は、nPIO7～0入力信号を、RR4、5レジスタで読み込んだときの遅延時間を示しています。IC内蔵フィルタは無効にしています。
右下図は、nPIO7～0 出力信号データを、WR4,5 レジスタに書き込んだときの遅延時間を示しています。



記号	項　　目	最小	最大	単位
tDI	入力信号 → データ 遅延時間		17	nS
tDO	WRN↑→ データセットアップ時間		23	nS

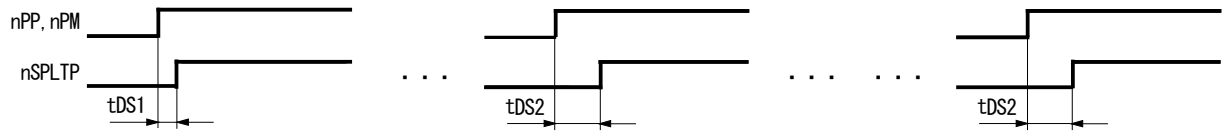
10.2.6 スプリットパルス

スプリットパルスが開始されるドライブパルスの立ち上がりからスプリットパルスが Hi になるまでの遅延時間です(スプリットパルスは正論理としています)。

開始パルス有りの場合には、第 1 スプリットパルスだけがドライブパルスと同時に出力されますが、第 2 スプリットパルス以降はドライブパルスより1CLK 遅れて出力されます。
開始パルス無しの場合には、すべてのスプリットパルスがドライブパルスより1CLK 遅れて出力されます。

■ スプリットパルスのモード設定で開始パルス有りの場合

スプリットパルスのモード設定で開始パルス有りを設定した時の開始ドライブパルス立ち上がり時からスプリットパルスが Hi になるまでの遅延時間です。
tDS1 は第 1 スプリットパルスの遅延時間です。tDS2 は第 2 スプリットパルス以降の遅延時間を示しています。第 2 パルス以降は 1CLK 遅れます。

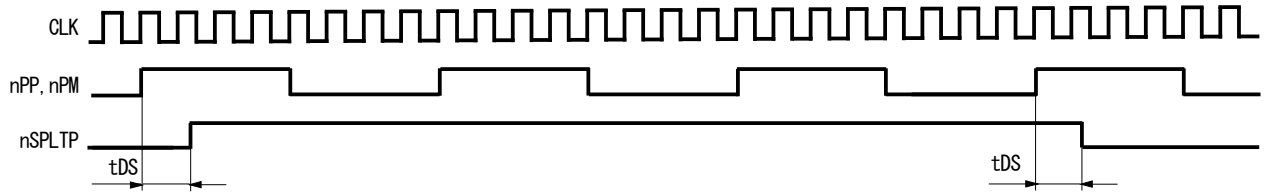


記号	項　　目	最小	最大	単位
tDS1	nPP, nPM↑ → nSPLTP↑ 遅延時間		20	nS
tDS2	nPP, nPM↑ → nSPLTP↑ 遅延時間		tCYC +20	nS

tCYC は CLK の周期です。

■ スプリットパルスのモード設定で開始パルス無しの場合

スプリットパルスのモード設定で開始パルス無しを設定した時の開始ドライブパルス立ち上がり時からスプリットパルスが Hi になるまでの遅延時間です。

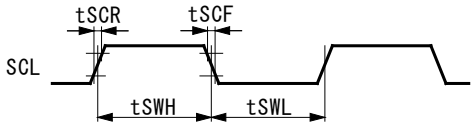


記号	項　　目	最小	最大	単位
tDS	nPP, nPM↑ → nSPLTP↑ 遅延時間		tCYC +20	nS

tCYC は CLK の周期です。

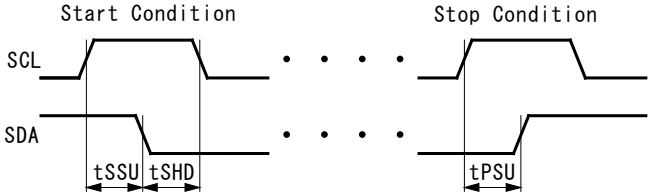
10.2.7 I²C シリアルバス

■ SCL クロック



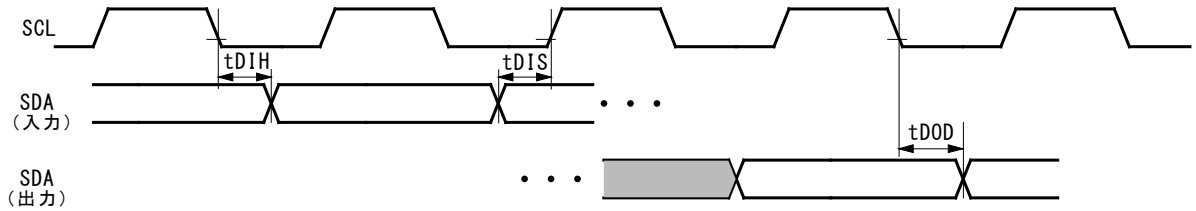
記号	項　　目	最小	最大	単位
fSCL	SCL クロック周波数		400	KHz
tSWH	SCL クロック Hi レベル幅	600		nS
tSWL	SCL クロック Low レベル幅	1300		nS
tSCR	SCL クロック立ち上がり時間		300	nS
tSCF	SCL クロック立ち下がり時間		300	nS

■ スタート/ストップコンデション



記号	項　　目	最小	最大	単位
tSSU	スタートコンデションセットアップ時間	600		nS
tSHD	スタートコンデションホールド時間	600		nS
tPSU	ストップコンデションセットアップ時間	600		nS

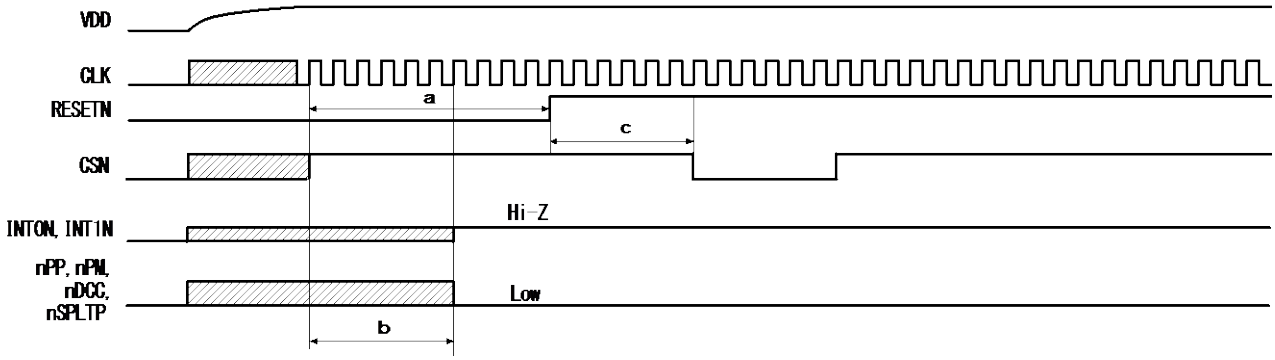
■ 書き込み/読み出し SDA データ



記号	項　　目	最小	最大	単位
tDIH	SDA 入力ホールド時間	0		nS
tDIS	SDA 入力セットアップ時間	100		nS
tDOD	SDA 出力遅延時間	0	900	nS

11. 入出力信号タイミング

11.1 パワーオンタイミング

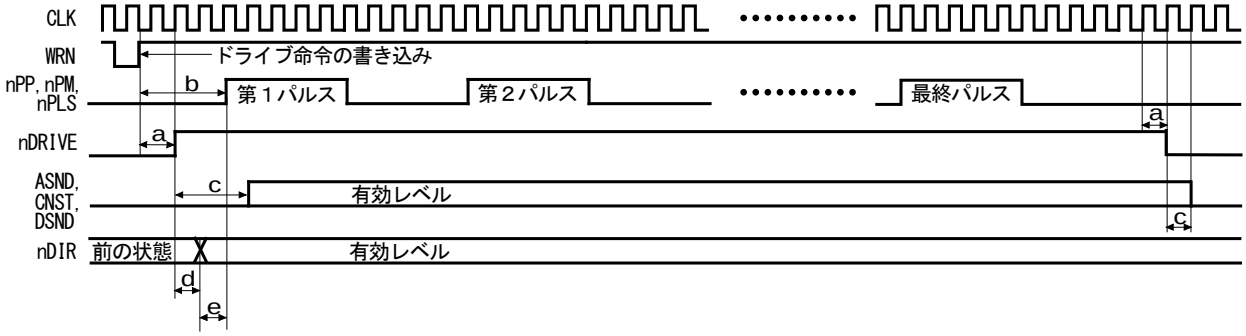


a. リセット入力信号(RESETN)は、CLK入力後、CLK×8サイクル以上Lowレベルであることが必要です。

b. 電源投入時の出力信号は、RESETNがLowレベルであり、CLKが入力されている状態で、最大でCLK×6サイクル後に、上図に示すレベルに確定します。

c. RESETNがHiレベルに上がってから、最大でCLK×4サイクルの間は、本ICへのリード／ライトはできません。

11.2 ドライブ開始／終了時



a. ドライブ状態出力信号(nDRIVE)は、ドライブ命令が書き込まれた時、WRNの↑から最大CLK2サイクル後にHiレベルになり、最終パルスのLow期間後からCLK1サイクル後にLowレベルに戻ります。

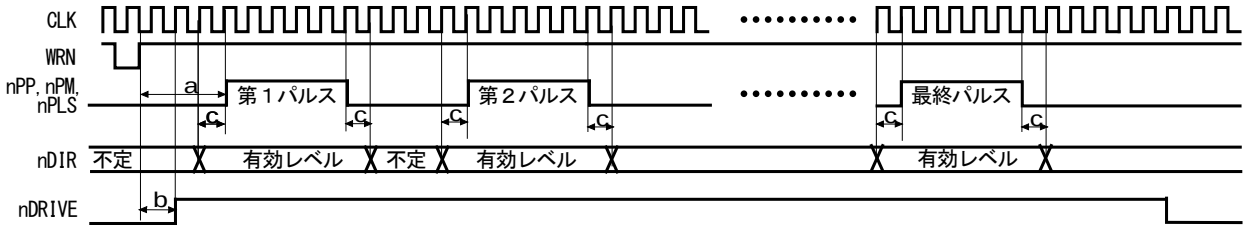
b. ドライブパルス(nPP, nPM, nPLS)は、本図では正パルスの場合を示しています。ドライブ命令が書き込まれた時、WRNの↑から最大CLK4サイクル後に第1パルスが出力されます。

c. ASND,CNST,DSNDは、nDRIVEの↑からCLK3サイクル後に有効レベルになり、DRIVEの↓からCLK1サイクル後にLowレベルに戻ります。

d. ドライブ出力パルス方式を1パルス方式に設定したときのnDIR(方向)信号は、nDRIVEの↑からCLK1サイクル後に有効レベルに変化します。ドライブ終了後も次のドライブ命令が書き込まれるまでそのレベルを保持します。

e. nDIR(方向)信号が有効レベルに変化してからCLK1サイクル後にドライブパルス(nPLS)の第1パルスが出力されます。

11.3 補間ドライブ開始

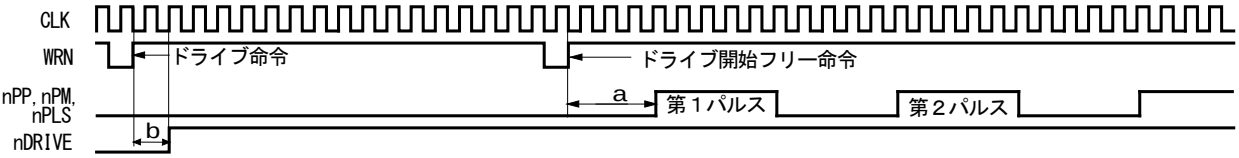


a. 補間ドライブ時のドライブパルス(nPP, nPM, nPLS)は、ドライブ命令が書き込まれた時、WRNの↑から最大CLK4サイクル後に第1パルスが出力されます。

b. nDRIVEは、WRNの↑から最大CLK2サイクル後にHiレベルになります。

c. ドライブ出力パルス方式を1パルス方式に設定したときの nDIR(方向)信号は、補間ドライブ時、ドライブパルスの Hi レベル幅とその前後 CLK1サイクルの間、有効レベルになります。(ドライブパルス:正論理パルスのとき)

11.4 ドライブ開始フリー

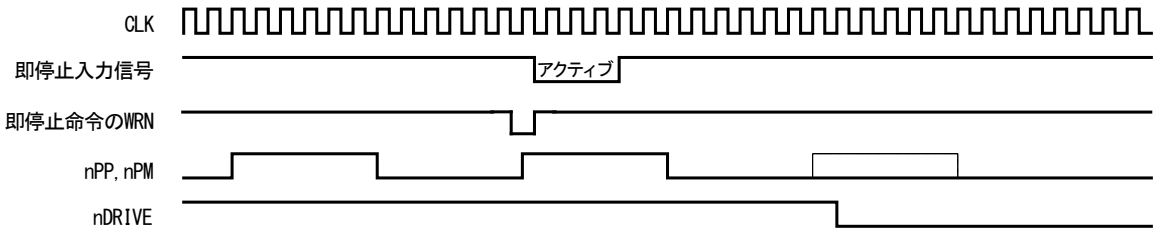


a. 各軸のドライブパルス(nPP, nPM, nPLS)は、ドライブ開始フリー命令が書き込まれた時、WRNの↑から最大CLK4サイクル後に第1パルスが出力されます。

b. nDRIVE は、各軸のドライブ命令書き込み時、WRN の↑から最大 CLK2サイクル後に Hi レベルになります。

11.5 ドライブ即停止

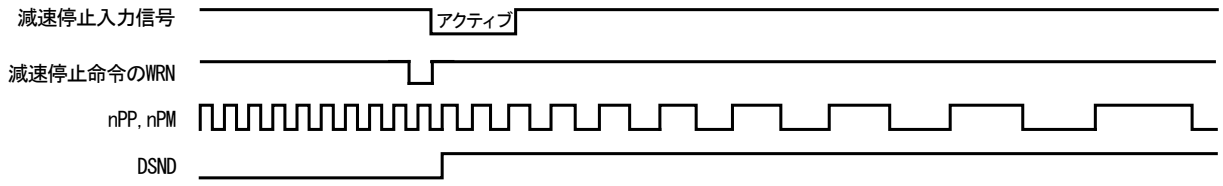
即停止入力信号と、即停止命令の動作タイミングです。即停止入力信号は、EMGN、nLMTP/M(即停止モードに設定時)、nALARM、です。
即停止入力信号がアクティブレベルになると、または即停止命令が書き込まれると、現在出力中のドライブパルスを出力したのちに、パルス出力を停止します。



即停止入力信号は、入力信号フィルタを無効にしている場合でも、CLK2サイクル以上のパルス幅が必要です。
入力信号フィルタを有効にすると、フィルタの時定数の値に応じて入力信号は遅延します。

11.6 ドライブ減速停止

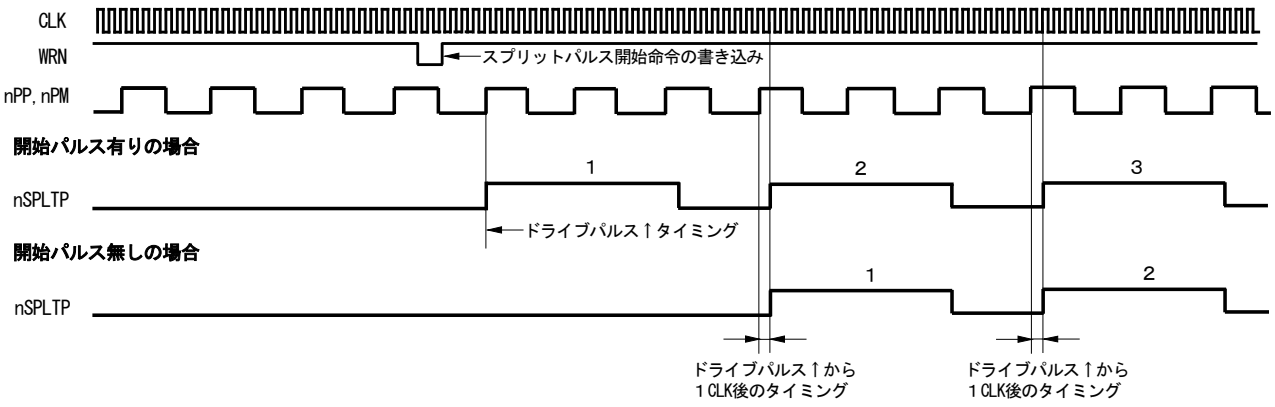
減速停止入力信号と、減速停止命令の動作タイミングです。減速停止入力信号は、nSTOP2～0、nLMTP/M(減速停止モードに設定時)です。
減速停止入力信号がアクティブレベルになると、または減速停止命令が書き込まれると、現在出力中のドライブパルスを出力したのちに、減速に移行します。



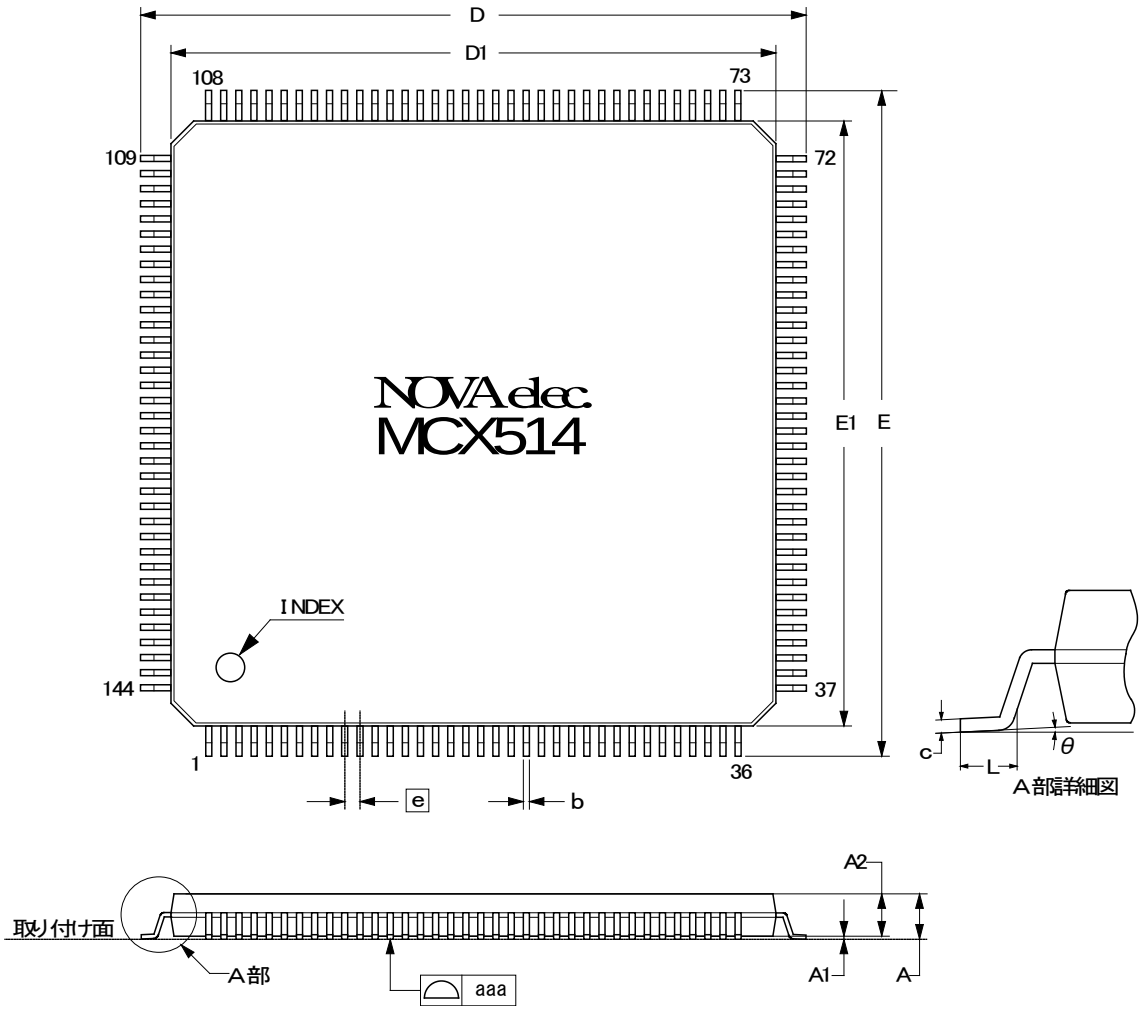
入力信号フィルタを有効にすると、フィルタの時定数の値に応じて入力信号は遅延します。

11.7 スプリットパルスの詳細タイミング

スプリットパルスの動作モードで開始パルスを有りに設定した場合、第1スプリットパルスのみ、ドライブパルス↑のタイミングでスプリットパルスがHiレベルになります。第2スプリットパルス以降は、ドライブパルス↑から1CLK後にスプリットパルスがHiレベルになります。そのため、第1スプリットパルスのHiレベル幅が第2スプリットパルス以降に比べ1CLK分、長くなります。
動作モードで開始パルス無しに設定した場合は、全てのスプリットパルスは、ドライブパルス↑から1CLK後にスプリットパルスがHiレベルになります(正論理パルス設定時)。



12. 外形寸法



パッケージサイズ　20×20×1.4 mm

記号	寸法 mm			説　明
	最小	標準	最大	
A	―	―	1.7	取り付け面からパッケージ本体最上端部までの高さ
A1	―	0.1	―	取り付け面からパッケージ本体下端までの高さ
A2	―	1.4	―	パッケージ本体の上端から下端までの高さ
b	0.17	―	0.27	端子の幅
c	0.09	―	0.2	端子の厚さ
D	21.8	22	22.2	端子を含むパッケージ長さ方向の最大長
D1	19.8	20	20.2	端子を除くパッケージ本体の長さ
E	21.8	22	22.2	端子を含むパッケージ幅方向の最大長
E1	19.8	20	20.2	端子を除くパッケージ本体の幅
e	0.5			端子ピッチ基準寸法
L	0.3	―	0.75	取り付け面に接触する端子の平たん部長さ
θ	0°	―	10°	取り付け面に対する端子平たん部角度
aaa	0.08			端子最下面の均一性（垂直方向の許容値）

13. 保管条件と推奨実装条件

13.1 本 IC の保管について

本ICの保管に際しては以下の項目に対してご注意願います。
(1) 投げたり落としたりしないでください。包装材が破れて気密性が損なわれる場合があります。
(2) 保管は、防湿梱包未開封の状態で30℃、85%RH 以下の環境とし、12ヶ月以内にご使用下さい。
(3) 有効期限が過ぎた場合には、排湿処理として125℃±5℃で20時間以上36時間以内のベーキングを実施してください。ベーキング回数は、2回以内としてください。また、有効期限内においても防湿梱包の気密が損なわれた場合には排湿処理を行ってください。
(4) 排湿処理の実施に際しては、静電気によるデバイスの破壊防止を行って下さい。
(5) 防湿梱包開封後は、30℃、70%RH 以下の環境条件下で保管し、7日以内での実装をお願いします。なお、上記許容放置期間を過ぎたICにつきましては、実装前に必ずベーキング処理を実施願います。

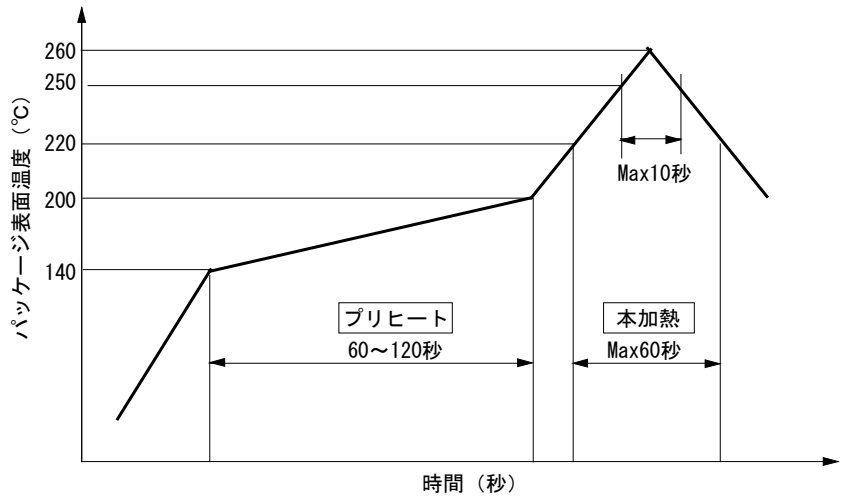
13.2 はんだごてによる標準実装条件

本ICのはんだごてによる標準実装条件は、以下の通りです。
(1) 実装方法: はんだごて(端子部分加熱)
(2) 実装条件: 端子温度:350℃以下、 時間:5秒以内、回数:2回以下

13.3 リフローによる標準実装条件

本ICのリフローによる標準実装条件は、以下の通りです。

実装方法	(1) 赤外線 (2) 温風 (3) 赤外線・温風併用
最高リフロー温度(パッケージ表面温度)	260℃以下
250℃以上の時間	10秒以内
220℃以上の時間	60秒以内
プリヒートの温度140℃～200℃の時間	60～120秒
リフロー回数	2回以内

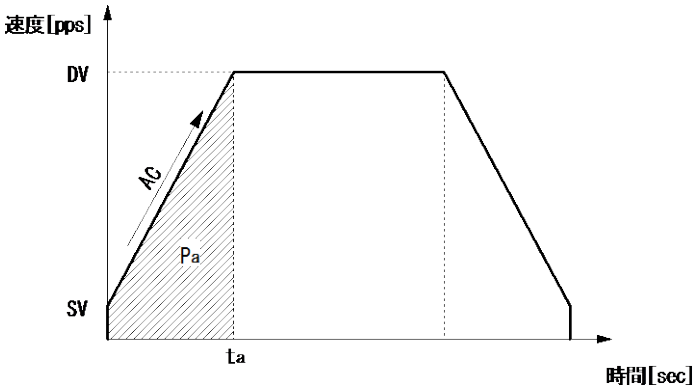


MCX514リフロー実装条件

付録A　加減速ドライブのための計算式

A-1 直線加減速ドライブの場合

（ただしCLK = 16MHz）



DV：ドライブ速度[pps]
SV：初速度[pps]
AC：加速度[pps/sec]
ta：加速時間[sec]
Pa：加速消費パルス数

◎ 初速度 SV、ドライブ速度 DV、加速時間 ta が与えられたときの、加速度 AC 計算式

$$\text{加速度}\quad AC = \frac{DV - SV}{ta} \quad [\text{pps/sec}]$$

◎ 初速度 SV、ドライブ速度 DV、加速度 AC が与えられたときの、加速時間 ta 計算式

$$\text{加速時間}\quad ta = \frac{DV - SV}{AC} \quad [\text{sec}]$$

◎ 初速度 SV、ドライブ速度 DV、加速度 AC が与えられたときの、加速消費パルス数 Pa 計算式

$$\text{加速消費パルス数}\quad Pa = \frac{DV^2 - SV^2}{2 \times AC}$$

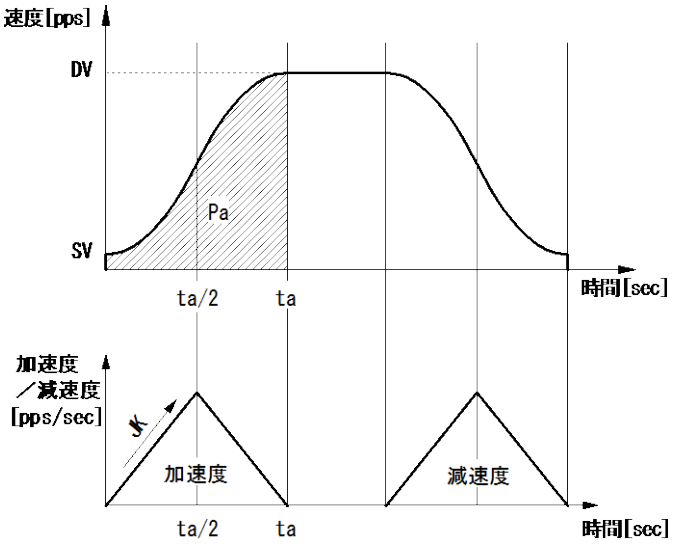
加速度 AC、加速時間 ta、加速消費パルス数 Pa を、減速度 DC、減速時間 td、減速消費パルス数 Pd に置き換えることで、それぞれ **減速度 DC、減速時間 td、減速消費パルス数 Pd** を求めることが出来ます。

【注意】

- 上記の計算式は理想的な式であり、実際の IC の動作においては多少の差異が生じます。

A-2 S 字加減速ドライブの場合

（ただし CLK = 16MHz）



DV：ドライブ速度[pps]
SV：初速度[pps]
JK：加速度増加率[pps/sec^2]
ta：加速時間[sec]
Pa：加速消費パルス数

加速度 AC は 1FFF FFFFh 固定です。

◎ 初速度 SV、ドライブ速度 DV、加速時間 ta が与えられたときの、加速度増加率 JK 計算式

$$\text{加速度増加率}\quad JK = \frac{4(DV - SV)}{ta^2} \quad [\text{pps/sec}^2]$$

◎ 初速度 SV、ドライブ速度 DV、加速度増加率 JK が与えられたときの、加速時間 ta 計算式

$$\text{加速時間}\quad ta = 2\sqrt{\frac{DV - SV}{JK}} \quad [\text{sec}]$$

◎ 初速度 SV、ドライブ速度 DV、加速度増加率 JK が与えられたときの、加速消費パルス数 Pa 計算式

$$\text{加速消費パルス数}\quad Pa = (DV + SV)\sqrt{\frac{DV - SV}{JK}}$$

加速度増加率 JK、加速時間 ta、加速消費パルス数 Pa を、減速度増加率 DJ、減速時間 td、減速消費パルス数 Pd に置き換えることで、それぞれ **減速度増加率 DJ、減速時間 td、減速消費パルス数 Pd** を求めることが出来ます。

【注意】

- 部分 S 字加減速では、上記計算式は成立しません。
- 上記の計算式は理想的な式であり、実際の IC の動作においては多少の差異が生じます。

付録B　入力クロックが 16MHz 以外のパラメータ計算式

MCX514 の入力クロック周波数を f_{CLK} (Hz)としたときの、各速度設定値、タイマー設定値の計算式を以下に示します。

初速度 [pps] $= SV \times \dfrac{f_{CLK}}{16\times10^6}$

ドライブ速度[pps] $= DV \times \dfrac{f_{CLK}}{16\times10^6}$

加速度[pps/sec] $= AC \times \left(\dfrac{f_{CLK}}{16\times10^6}\right)^2$

減速度[pps/sec] $= DC \times \left(\dfrac{f_{CLK}}{16\times10^6}\right)^2$

加速度増加率[pps/sec^2] $= JK \times \left(\dfrac{f_{CLK}}{16\times10^6}\right)^3$

減速度増加率[pps/sec^2] $= DJ \times \left(\dfrac{f_{CLK}}{16\times10^6}\right)^3$

原点検出速度[pps] $= HV \times \dfrac{f_{CLK}}{16\times10^6}$

速度増減値[pps] $= IV \times \dfrac{f_{CLK}}{16\times10^6}$

タイマー値[μsec] $= TM \times \dfrac{16\times10^6}{f_{CLK}}$

[パラメータ記号]

SV：初速度設定値
DV：ドライブ速度設定値
AC：加速度設定値
DC：減速度設定値
JK：加速度増加率設定値
DJ：減速度増加率設定値
HV：原点検出速度設定値
IV：速度増減設定値
TM：タイマー設定値

同期パルス出力幅(同期動作)、偏差カウンタクリア出力信号幅(自動原点出し)、ステップ間タイマー時間(自動原点出し)、入力信号遅延時間(入力信号フィルタ)についても、それぞれ $\dfrac{16\times10^6}{f_{CLK}}$ で補正が必要です。

付録C　MCX300 シリーズとの相違点

従来のMCX300シリーズと、MCX514との主な相違点を以下に示します。
機能の詳細については、取扱説明書の当該項目を参照してください。

	項目	MCX300シリーズ	MCX514
1	未使用入力端子の処理	オープン可。 （IC内部でVDDにプルアップされている）	IC内部でプルアップされていない入力端子がある。それらの未使用入力端子はGNDまたはVDDに接続する。詳細は5章を参照のこと。
2	リセット（RESETN）信号の幅	クロック4サイクル以上必要	クロック8サイクル以上必要
3	コマンドリセットの方法	WR0レジスタに8000h（D15ビット：1）を書き込む。	WR0レジスタに 00FFh を書き込む。
4	速度パラメータ値の設定	速度レンジ設定あり（倍率：1～500倍） 速度パラメータ値は実際の値と倍率によって設定する。	速度レンジ設定なし（速度レンジフリー） 速度パラメータ値は実際の値を設定する。
5	定量パルスドライブ	・ ＋方向定量パルスドライブ 出力パルス数を正の値で指定する。 実行すると＋方向へ指定パルス数分ドライブする。 ・ －方向定量パルスドライブ 出力パルス数を正の値で指定する。 実行すると－方向へ指定パルス数分ドライブする。	・ 相対位置ドライブ 出力パルス数を正の値で指定し実行すると、＋方向へ指定パルス数分ドライブする。 出力パルス数を負の値で指定し実行すると、－方向へ指定パルス数分ドライブする。 ・ 反相対位置ドライブ 出力パルス数を正の値で指定し実行すると、－方向へ指定パルス数分ドライブする。 MCX300シリーズの－方向定量パルスドライブに対応したドライブ命令。 ・ 絶対位置ドライブ ドライブ終点として、目標点の論理位置カウンタ値を指定し実行する。
6	RR2レジスタ／エラー情報表示（ソフトリミット、ハードリミット信号、サーボアラーム信号、緊急停止信号）	ドライブ停止時においても、エラー要因がアクティブになると該当するエラー情報ビットが1になり、エラー要因が解消されると該当するエラー情報ビットは0に戻る。	ドライブ中にエラー要因がアクティブになると、（またはドライブ開始時にエラー要因がアクティブであると、）該当するエラー情報ビットが1になり、エラー要因が解消されても1が保持される。 ドライブが停止している時はエラー要因がアクティブになってもエラーにならない。 エラー・終了ステータスクリア命令（79h）、または次のドライブ開始でRR2の全ビットが0に戻る。ただし、補間ドライブ中にエラーが発生した場合には、必ずエラー・終了ステータスクリア命令（79h）を発行が必要。
7	ハードリミット機能の有効／無効	ハードウェアリミット入力信号（nLMTP, nLMTM） （MCX305はLMT+, LMT−）の機能の無効設定は不可。	ハードウェアリミット入力信号（nLMTP, nLMTM）の機能の有効／無効選択可能。
8	ソフトリミット値の設定	コンペアレジスタ（COMP+, COMP−）にソフトリミット値を設定する。 このためコンペアレジスタをソフトリミットとして使用するとき、コンペアレジスタの他の機能は使用できない。	ソフトリミット専用レジスタ（SLMT+, SLMT−）にソフトリミット値を設定する。
9	ソフトリミット機能の停止モード	減速停止のみ	減速停止／即停止から選択可能
10	直線加減速の三角波形防止	リセット時：無効設定	リセット時：有効設定
11	加速カウンタオフセットの設定値	リセット時：8	リセット時：0
12	命令コードおよびモード設定ビット	－	MCX300シリーズと異なる。

13	補間ドライブ時の位置・速度パラメータ設定	補間ドライブを続けて行なう場合、前の補間ドライブと設定するパラメータの値が同じであれば、再設定する必要はない。	補間ドライブ命令発行前に、必ず位置・速度パラメータの設定が必要。前の補間ドライブの値と同一であっても必ず再設定する。
14	2軸線速一定の設定方法	モード設定：WR5／D8,D9に設定	モード設定：補間モード設定命令(2Ah) WR6／D6,D7に設定
		速度レンジ設定：第2軸の速度レンジに主軸のレンジ値を1.414倍した値をセット。	不要